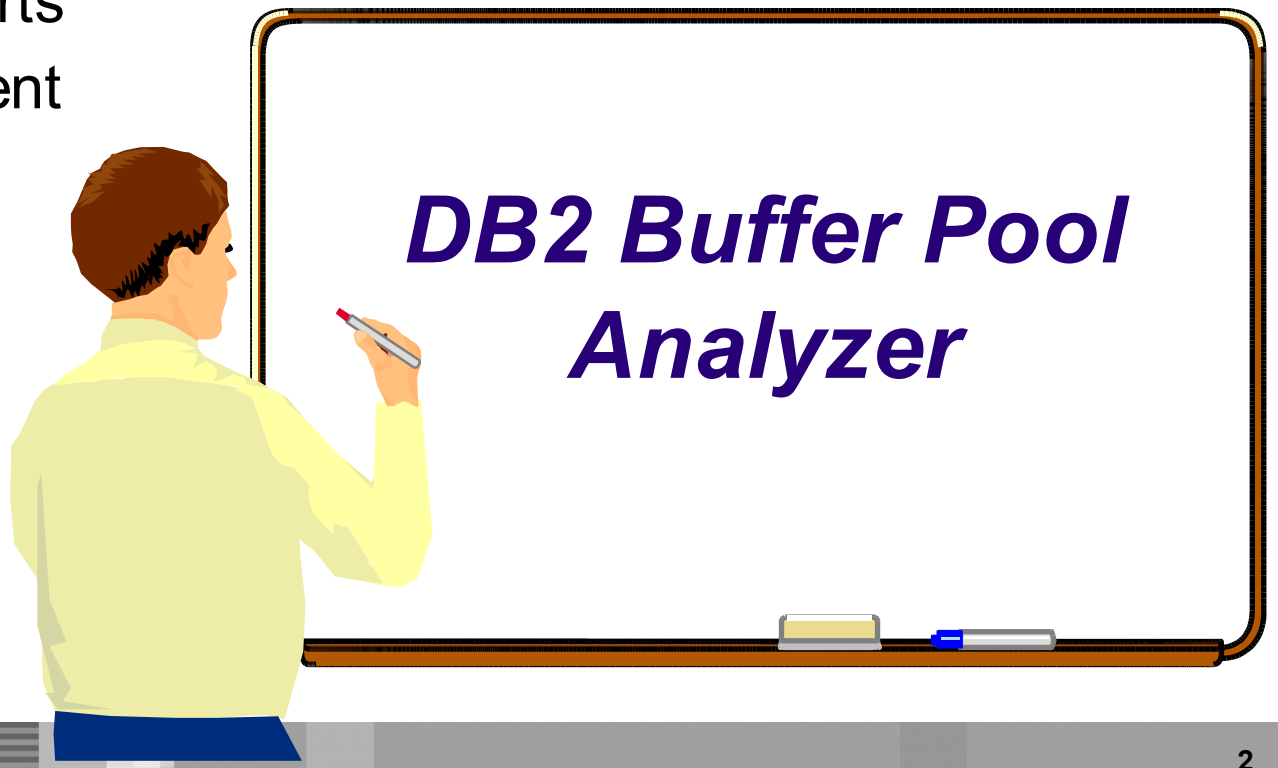
IBM Software Group

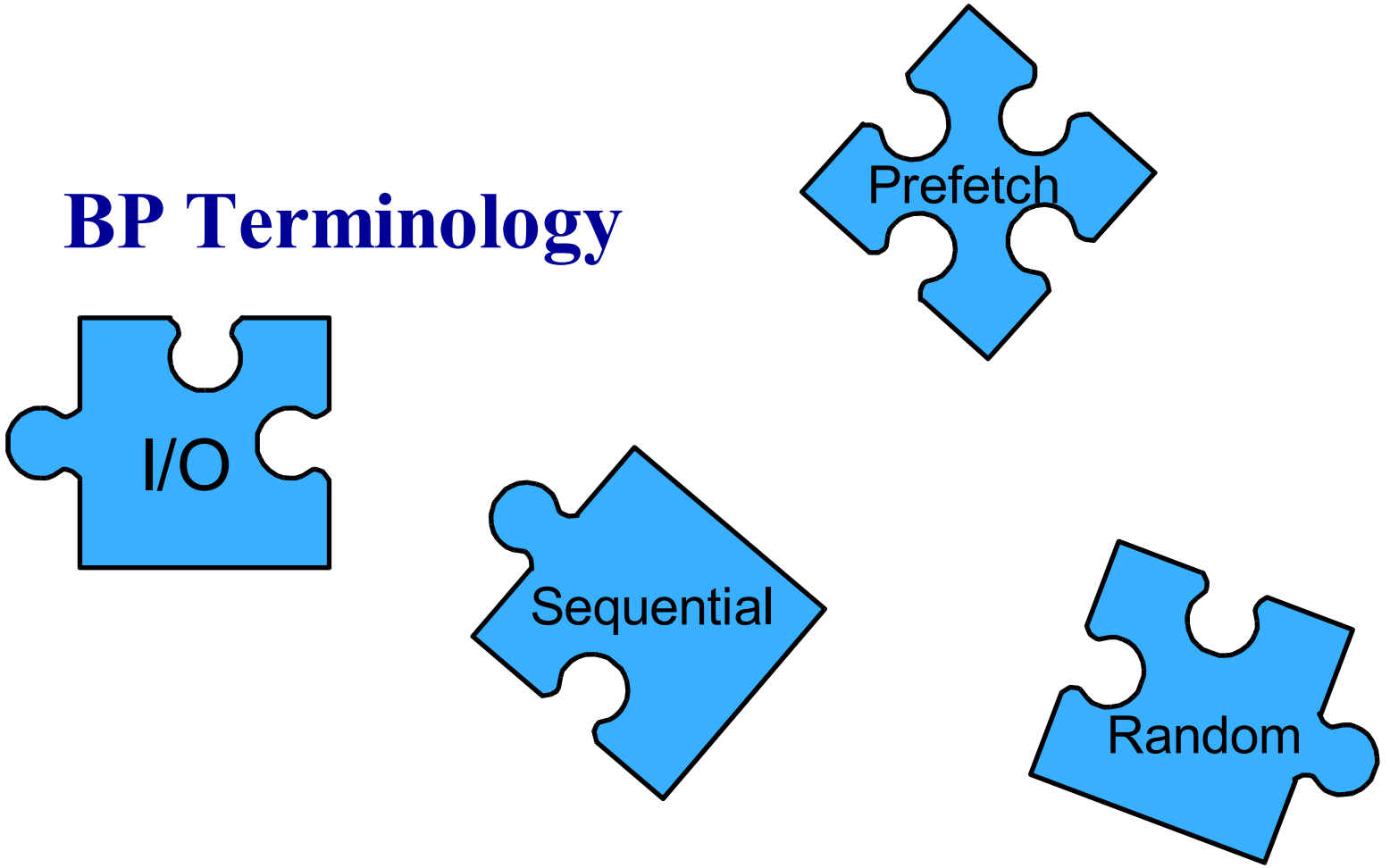
DB2 BUFFER POOL ANALYZER

Course Objectives

- Buffer Pool Terminology
- Basic Buffer Pool Monitoring
- Buffer Pool Analyzer
  - Traces
  - Batch Reports
  - Graphical Reports
  - Object Placement
  - Simulation



BP Terminology



What is a Buffer Pool?

- Areas of virtual storage in which DB2 temporarily stores pages of table spaces of indexes
- Cache of memory that can be reused
- Assists in reducing system I/O

GETPAGE

- Operation whereby DB2 accesses a data page from the BP
- Does not always require a disk I/O operation
- Counts
  - Unconditional requests - both successful and unsuccessful
  - Conditional requests that are successful
- Higher % of requested often data in a BP means less I/O
- Fewer GETPAGEs means less I/O

Synchronous I/O

- Always one I/O for one page
- Random access
- Number of Synchronous Read I/O Operations performed for both applications and utilities
- SYNCHRONOUS WRITES The number of immediate writes for a data set. Immediate writes occur when
  - Any synchronous write is triggered
  - An immediate write threshold is reached
  - No deferred write engines are available

Asynchronous I/O

- Number of Asynchronous Read I/Os caused by the normal Sequential Prefetch (both application and utilities
  - Sequential access
  - This number represents the **number of I/O operations**, not the number of pages read
- Number of Asynchronous Write I/O operations performed by media manager to a direct access storage device
- Multiple pages per I/O

Prefetch

- **Sequential prefetch**
  - Brings many pages into the virtual buffer pool before they are required
  - Reads several pages with a single I/O operation
    - Max Pages per I/O
      - 4k : 8 - 32 pages
      - 8k : 4 - 16 pages
      - 16k : 2 - 8 pages
      - 32k : 0 - 4 pages
  - Load, Reorg & Recover can be twice as much
- **List prefetch** is used to prefetch data pages that are not contiguous (such as through non-clustered indexes)
  - Rid Retrieval, Rid Sort (by page), Data Retrieval, incremental IC
- **Dynamic prefetch** is the process that triggers sequential prefetch because of DB2's sequential detection facility

Hit Ratio

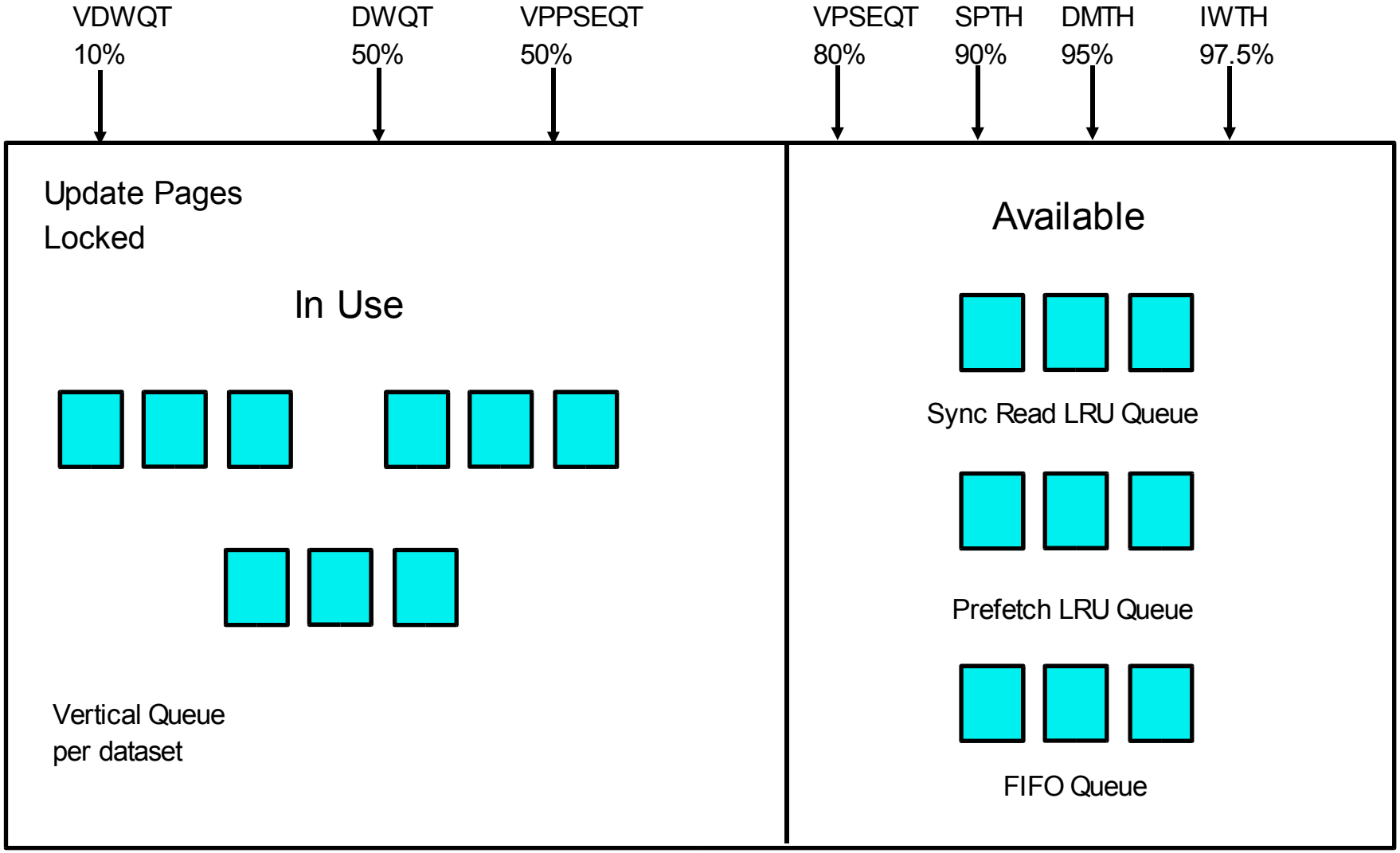
1- ((Sync I/O) + (Pref+ListPref+DynPref Pgs))/Getpages
or
(Getpages-Pages Rd)/Getpages * 100

- **A hit is any one requested page in the buffer pool**
  - Not an I/O
- **70% for data is good, > 90% better**
- **Indexes in a dedicated pool - mid 90% range (?)**
- **Application -** The # of getpage requests issued by applications satisfied by the BP, expressed as a % of all getpage requests issued by applications
- **System -** The # of getpage operations issued by DB2 and satisfied by the buffer pool, expressed as a % of all getpage requests
  - A negative value indicates that prefetched pages are not subsequently referenced. The reason for this is that the query stops before it reaches the end of the prefetched pages, or that the prefetched pages are stolen by DB2 for reuse before the query can access them

Buffer Pool Thresholds

- Preset values
- Represent a level of use - represented by a % of the total size of the BP
- When exceeded an action takes place
- Examples
  - BP shortage
  - Report normal management by DB2

Buffer Pool Thresholds



Relationship of thresholds and updated, inuse and available pages

Fixed Thresholds

- Cannot be altered
- If the thresholds are reached to often → increase the size of the BP
  - Do not want to reach these thresholds to often
  - ALTER BP size
  - Can have an impact on other BP's - depending on the total amount of available real storage
- More critical for performance than Variable Thresholds
- % of the BP that might be occupied by unavailable pages

Fixed Thresholds
IWTH - Immediate Write Threshold

- Default = 97.5%
- Checked whenever a page is to be updated
- If exceeded, the updated page is written to disk as soon as the update completes
  - Effects processor usage and I/O resource consumption
- The write is synchronous with the SQL request
  - The request waits until the write has been completed
  - The two operations do not occur concurrently
- Sometimes DB2 uses synchronous writes even when the IWTH has not been exceeded
  - Example > 2 checkpoints have passed w/o a page being written

Fixed Thresholds

DMTH - Data Management Threshold

- Default = 95%
- Maintained for each individual BP
- Checked before a page is read or updated
- If NOT exceeded, DB2 access the page in the BP once for each page, regardless of the # of rows are retrieved or updated for the page
- If the threshold has been exceeded, DB2 accesses the page in the virtual buffer pool once for each row that is retrieved or updated in that page
- In other words, retrieving or updating several rows in one page causes several page access operations
- Exceeding this threshold affects processor usage

Fixed Thresholds

SPTH - Sequential Prefetch Threshold

- Default = 90%
- Checked at 2 different times
  - Before scheduling a prefetch operation
    - If SPTH exceeded, the prefetch is not scheduled
  - During buffer allocation for an already scheduled prefetch operation
    - If SPTH is exceeded, the prefetch is canceled
- Operations that use sequential prefetch, such as those using large and frequent scans, are adversely affected

Thresholds You Can Change

-ALTER BUFFERPOOL Command

Changing a threshold for one BP does not change the threshold for any other BP

- VPSEQT - Sequential steal threshold
  - Default 80% (0 - 100%)
- HPSEQT - Hiperpool sequential steal threshold
  - Default 80% (0 - 100%) N/A for V8 and later
- VPPSEQT - Virtual buffer pool parallel sequential threshold
  - Default 50% (0 - 100%)
- VPXPSEQT - Virtual buffer pool assisting parallel threshold
  - Default 0% (0 - 100%)
- DWQT - Deferred write threshold
  - Default 50% (0 - 90%)
- VDWQT - Vertical deferred write threshold
  - Default 10% (0 - 90%) - Second integer for pages changed

Thresholds You Can Change

VPSEQT - Virtual Pool Sequential Threshold

- Default = 80% (range is 0 - 100%)
- % of virtual buffer pool that might be occupied by sequentially accessed pages
- The pages can be in the state updated ,in use ,or available
- Checked before stealing a buffer for a sequentially accessed page
- If VPSEQT = 0%,sequential pages cannot occupy space in the virtual buffer pool
  - Prefetch is disabled
  - Sequentially accessed pages are discarded when they are released.
- If the threshold is exceeded, DB2 tries to steal a buffer that holds a sequentially accessed page rather than one that holds a randomly accessed page.
- If VPSEQT = 100%, sequential pages can monopolize the entire virtual buffer pool

Thresholds You Can Change

HPSEQT - Hiperpool Sequential Steal Threshold

N/A for V8 and later

- Hiperpool Sequential Steal Threshold (HPSEQT)
- % of the hiperpool that might be occupied by sequentially accessed pages
- The effect of this threshold on the hiperpool is nearly identical to the sequential steal threshold on the virtual pool
- Note that changed pages are not written to the hiperpool; therefore, HPSEQT is the only threshold for hiperpools

Thresholds You Can Change
DWQT - Deferred Write Threshold

- % of virtual buffer pool that might be occupied by unavailable pages, including updated pages and pages in use
- Default = 50% (range 0 - 90%)
- Checked when an update to a page is complete
  - If % of unavailable pages > DWQT
    - Write operations are scheduled for up to 128 pages per data set to decrease the number of unavailable buffers to 10% below D WQT
    - For example, if DWQT is 50%,the number of unavailable buffers is reduced to 40%
- When the limit of DWQT is reached, the data sets containing the oldest updated pages are written asynchronously
- DB2 continues to write pages until the ratio goes below the DWQT

Thresholds You Can Change

VDWQT - Vertical Deferred Write Threshold

- Similar to the Deferred Write Threshold, but it applies to the number of **updated pages** for **one single page set** in the buffer pool
- If percentage or number of updated pages for the data set > VDWQT, writes up to 128 pages are scheduled for that data set
- You can specify VDWQT in one of the following ways:
  - % of the virtual buffer pool that might be occupied by updated pages from one single page set - Default = 10% (range 0 - 90%)
    - To use the % of the BP as the threshold, set the number of buffers to 0
  - Total number of buffers in the virtual buffer pool that might be occupied by updated pages from one single page set. Range from 0 to 9999
    - To use the number of buffers as the threshold, set the % threshold to 0
- The threshold is overridden by specific DB2 utilities that use a constant limit of 64 pages rather than a percentage of the virtual buffer pool size
- LOAD, REORG, and RECOVER use a constant limit of 128 pages
- VDWQT threshold cannot be > DWQT threshold

DB2 V8 VDWQT of 0 defaults to 1% of BP to avoid synchronous writes

Thresholds You Can Change

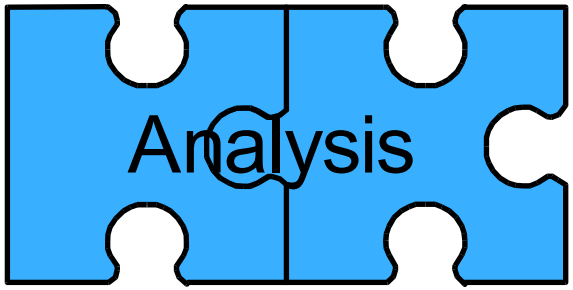
VPPSEQT - Virtual Buffer Pool Parallel Sequential Threshold

- This threshold is a part of the virtual buffer pool that might support parallel operations
- Measured as a % of the Sequential Steal threshold (VPSEQT)
- Default = 50% of the VPSEQT (range 0 - 100%)
- Setting VPPSEQT to zero disables parallel operation

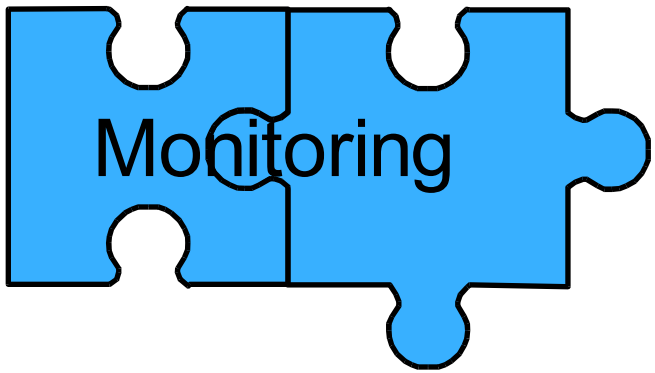
Thresholds You Can Change

VPXPSEQT - Virtual Buffer Pool Assisting Parallel Sequential Threshold

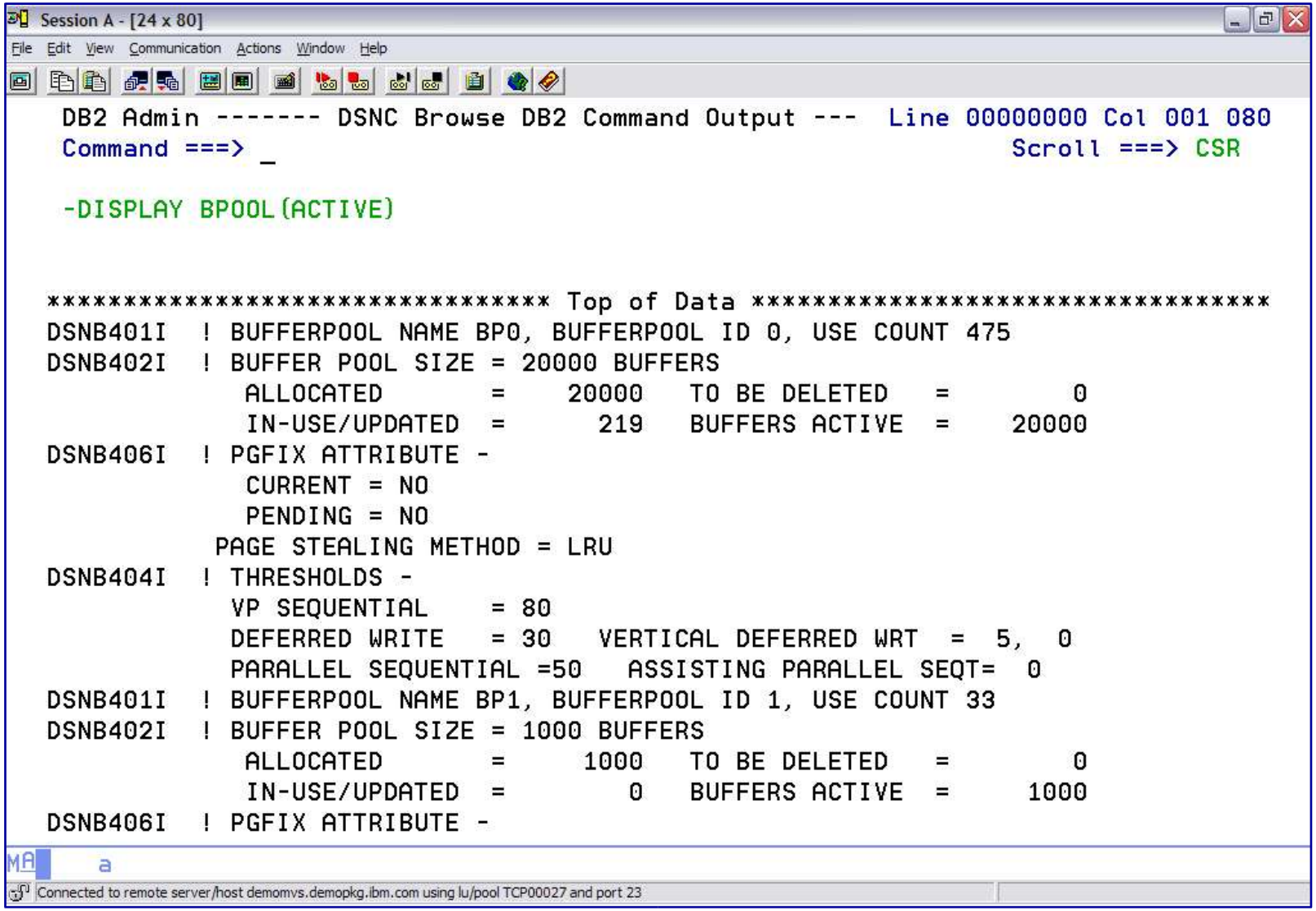
- This threshold is a part of the virtual buffer pool that might support parallel operations initiated from **another** DB2 in the data sharing group
- Measured as a % of VPPSEQT
- Default = 0% of VPPSEQT (range 0 - 100%)
- Setting VPXPSEQT to zero prevents DB2 from supporting Sysplex query parallelism at run time for queries that use this buffer pool



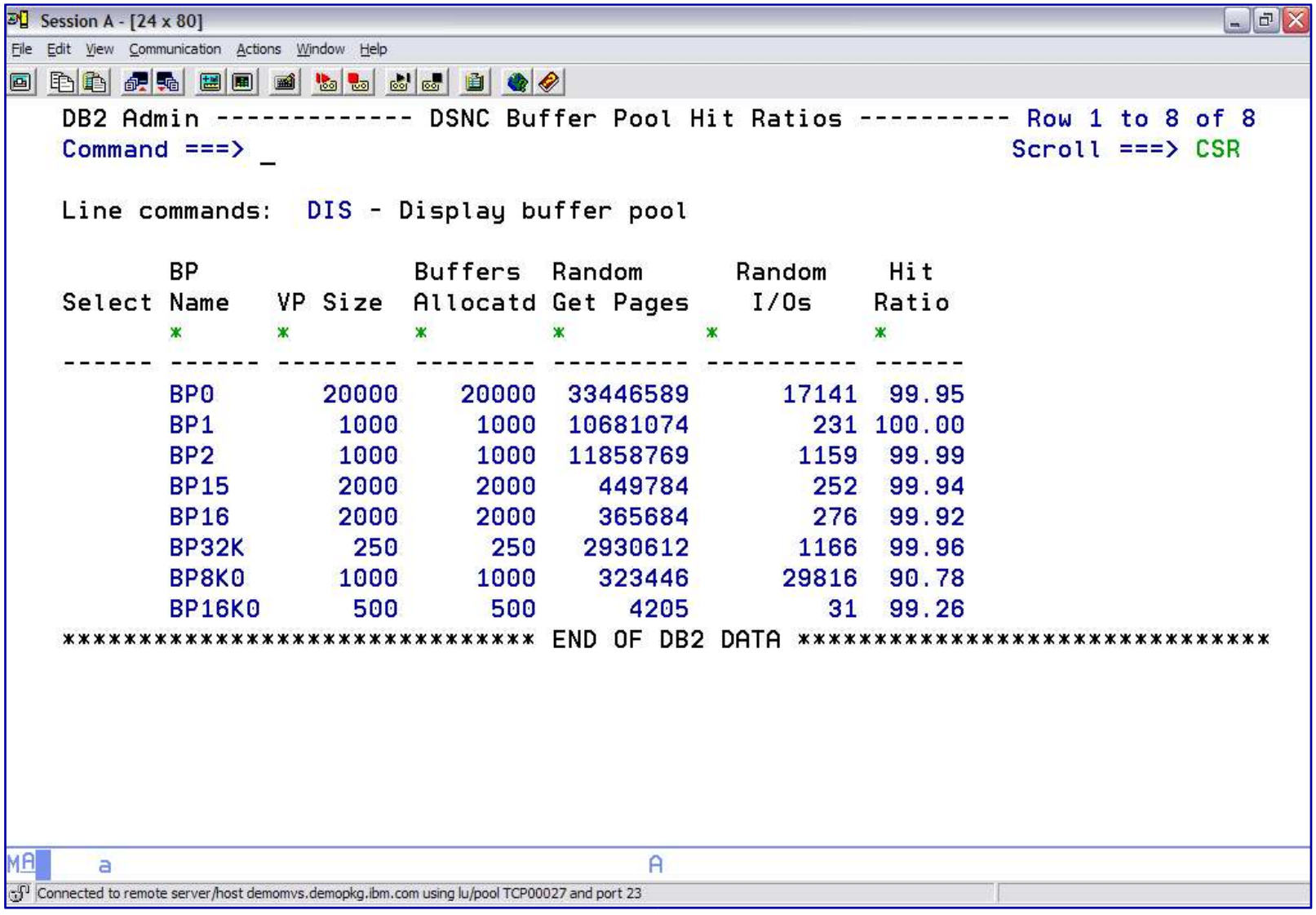
Basic Buffer Pool Monitoring



-DISPLAY BUFFERPOOL

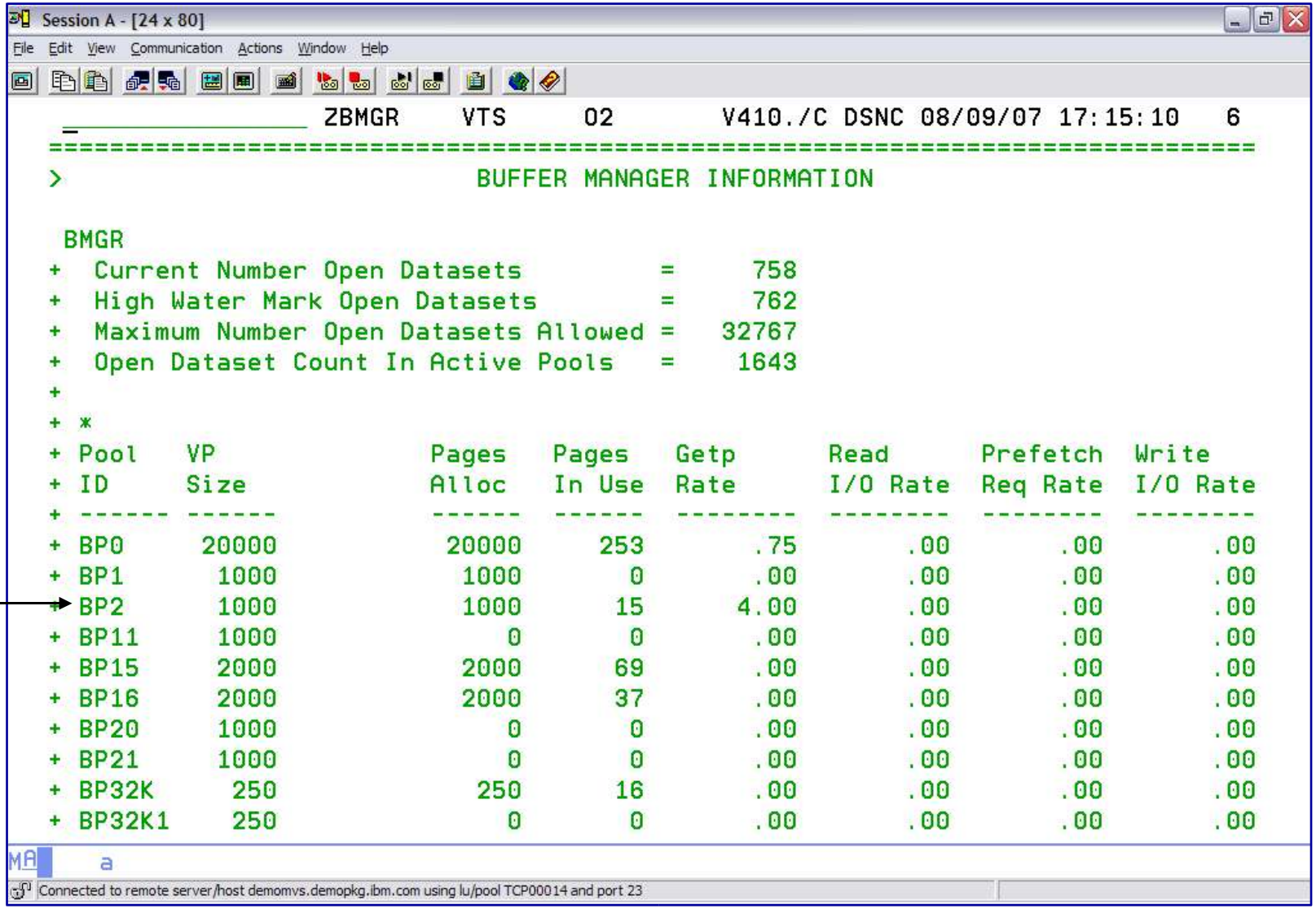


-DISPLAY BUFFERPOOL

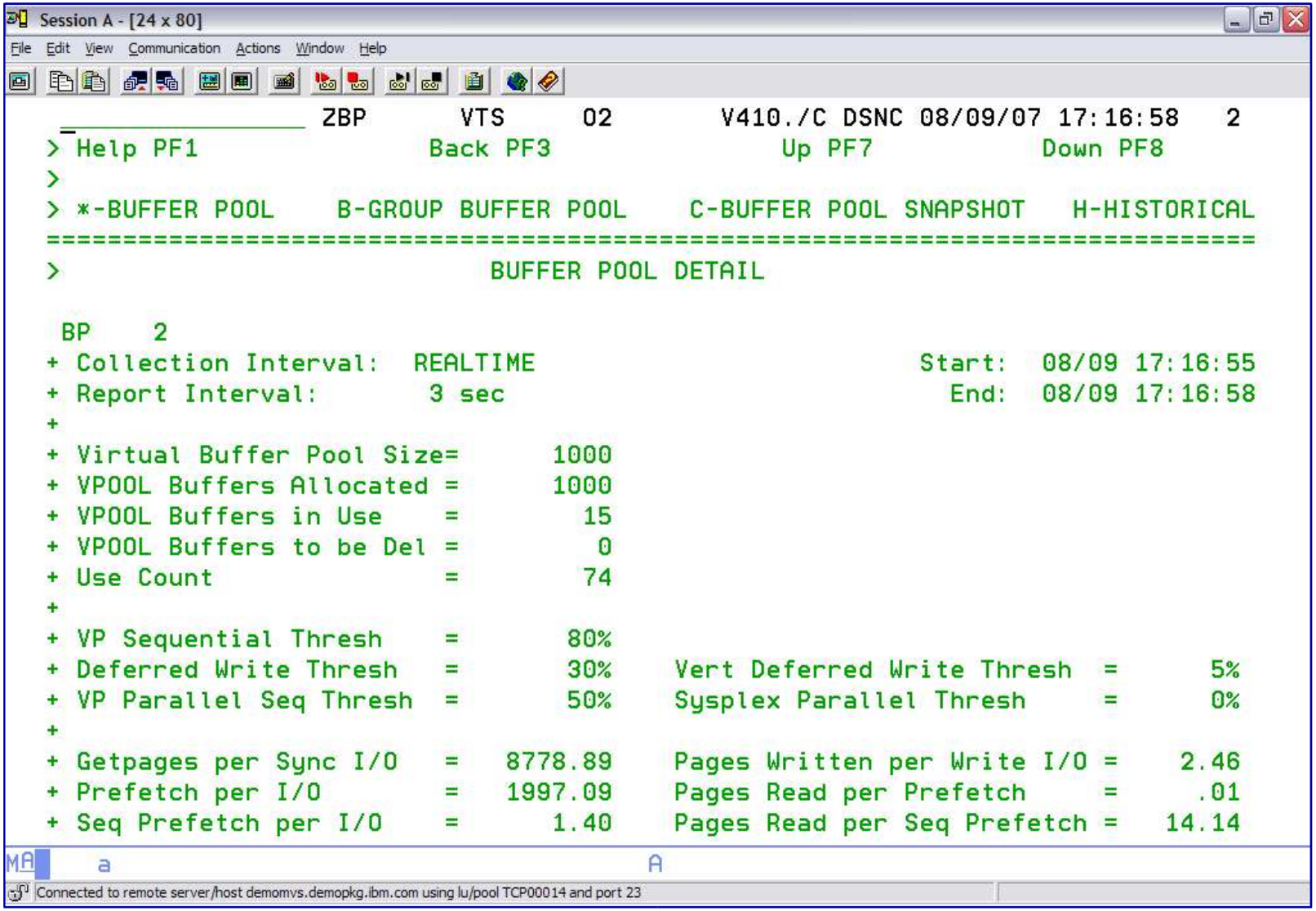


OMPE Buffer Manager Information

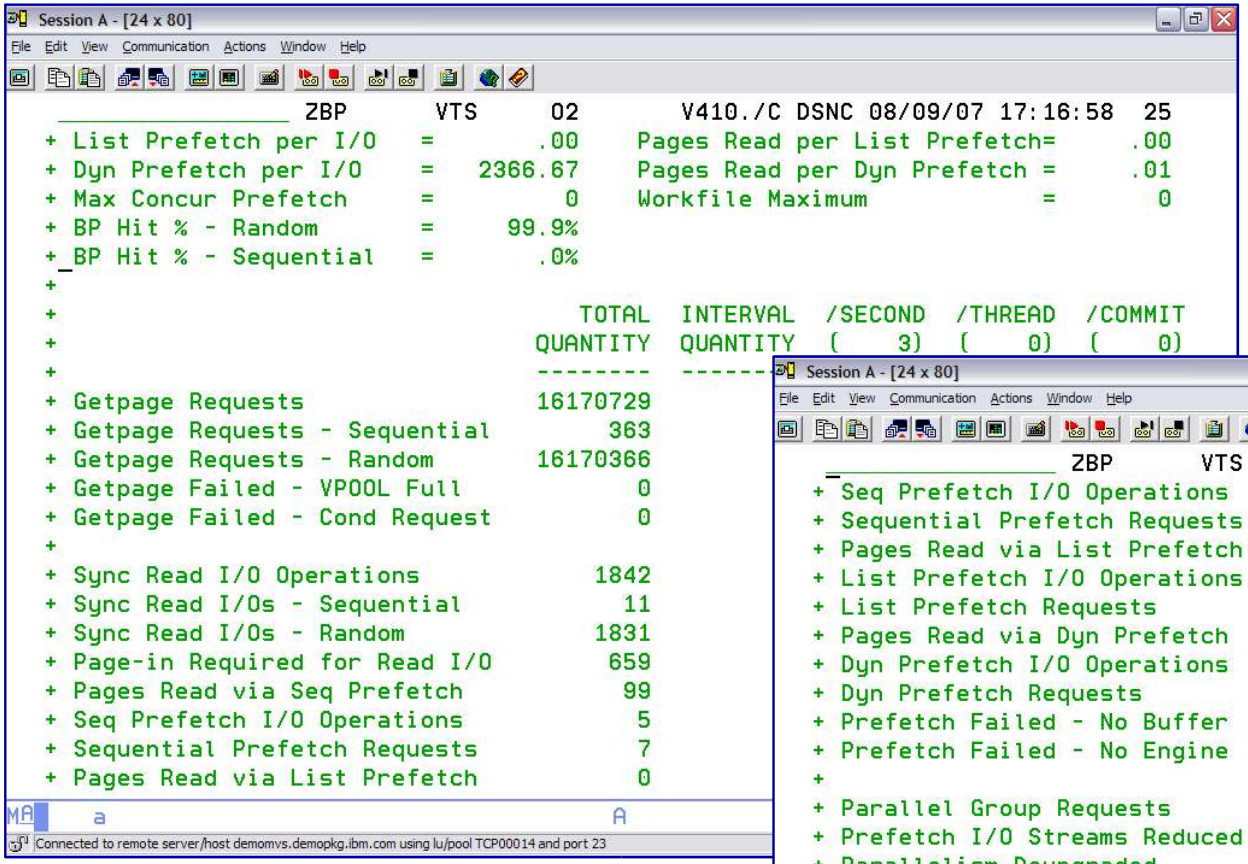
Put cursor on BP of interest and hit PF11

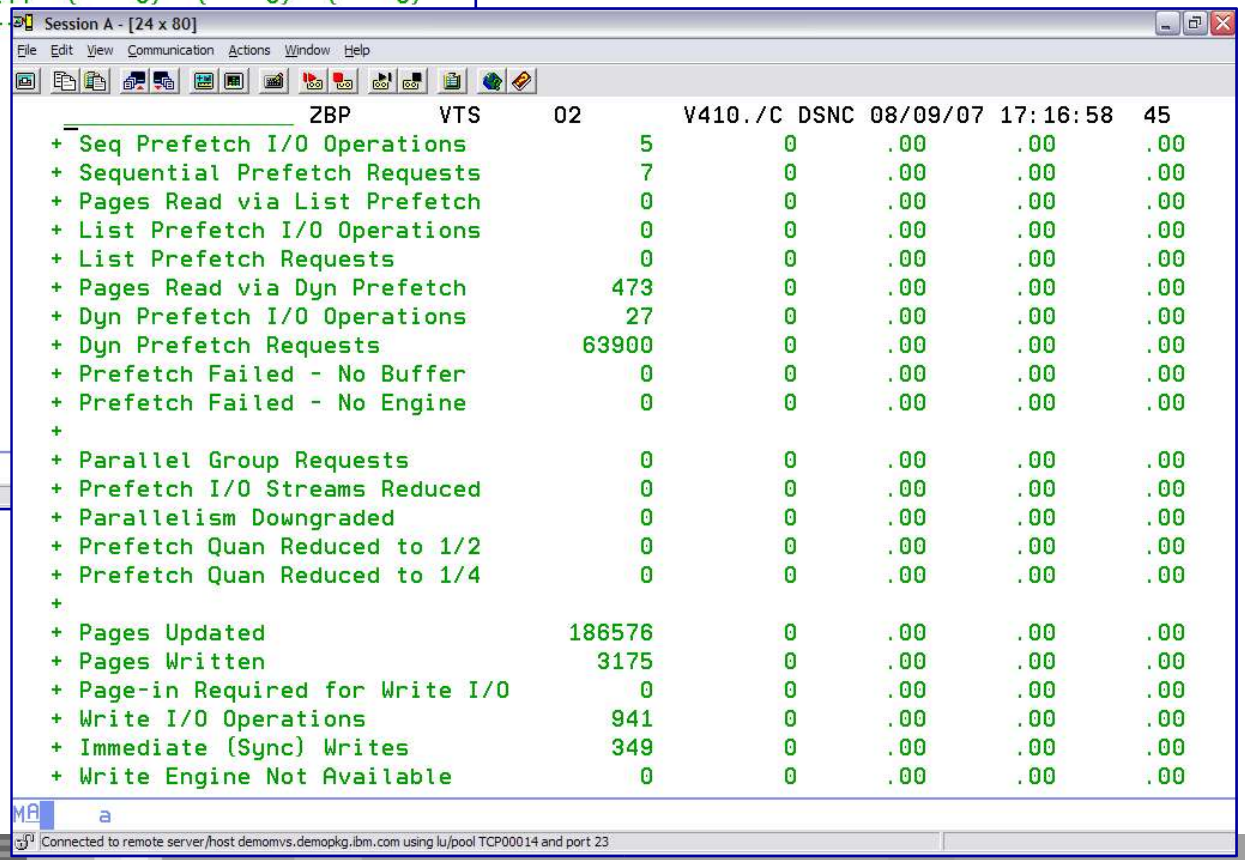


OMPE Buffer Pool Detail

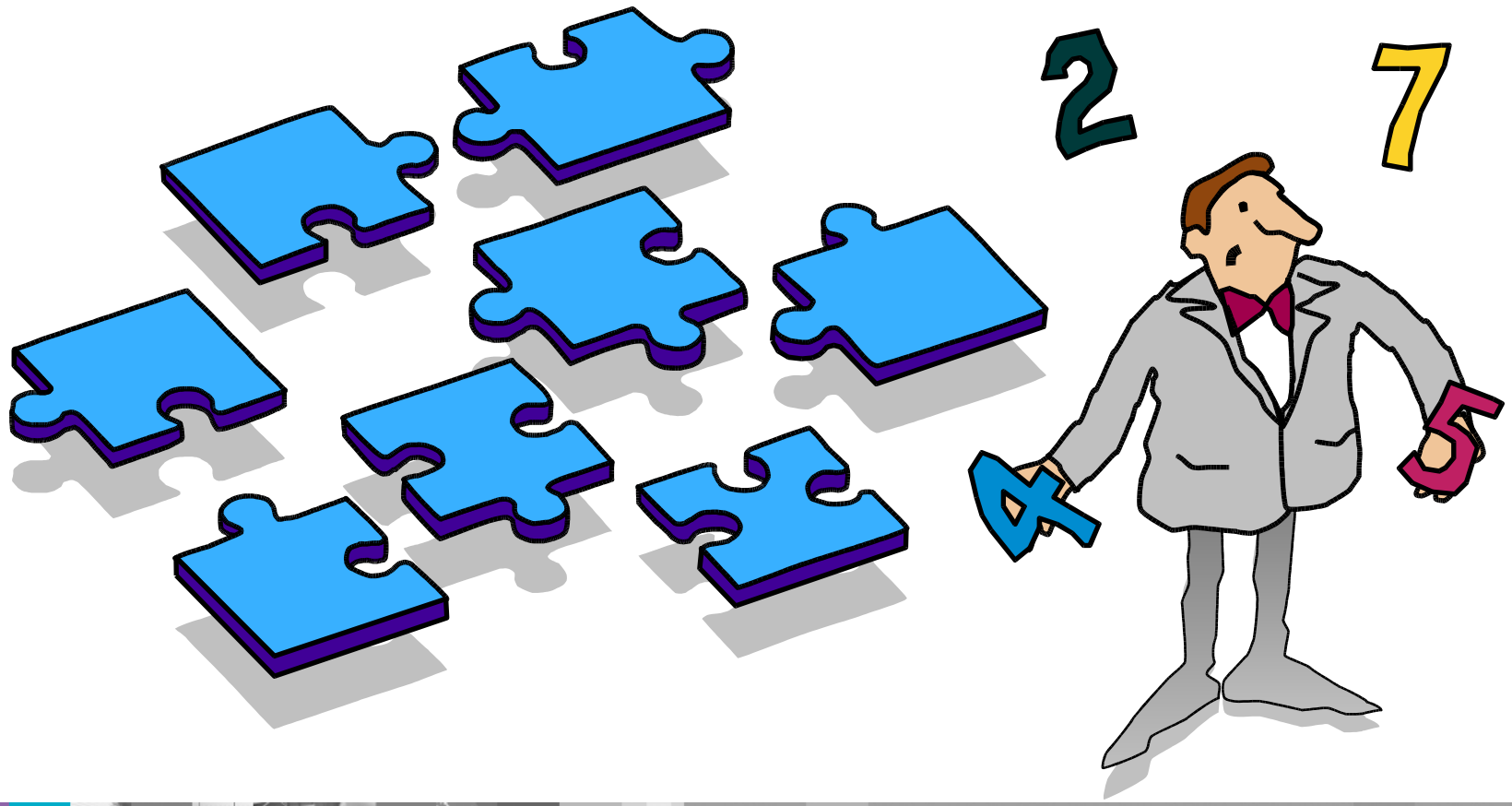


OMPE Buffer Pool Detail





Buffer Pool Analyzer



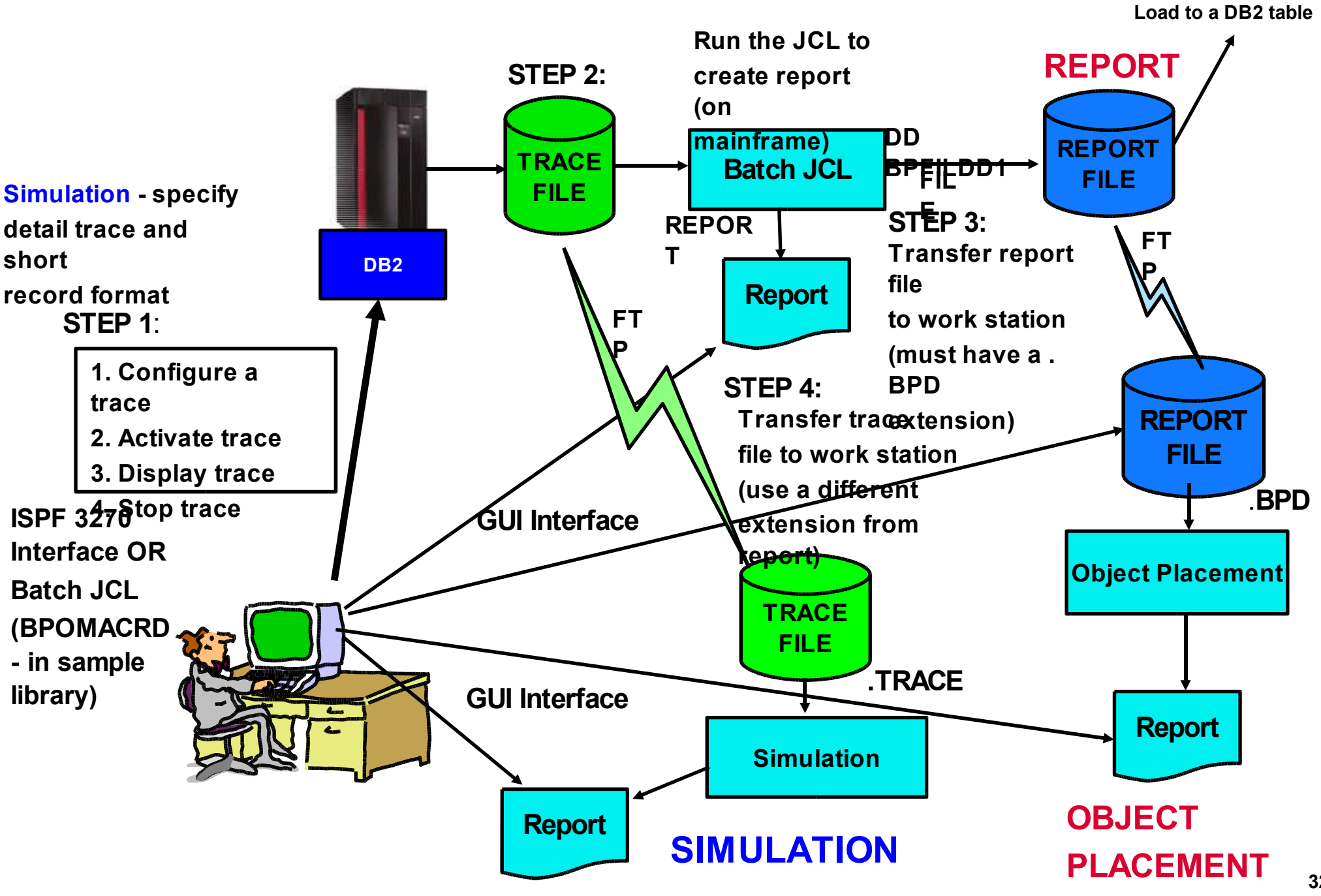
Buffer Pool Analyzer Features

- Collect buffer pool related performance data using batch JCL or Collect Report Data (ISPF)
- Create host-based reports on buffer pool and group buffer pool performance
- Convert performance data to formats suitable for loading into the performance database
- Display buffer pool performance data graphically
- Optimize buffer pool attributes including object assignment and buffer pool sizing
- Simulate the effects of different buffer pool sizes
- Perform long-term trend analysis of historical and current performance data

Buffer Pool Analyzer Benefits

- Easy monitoring of buffer pools and group buffer pools
  - Detect bottlenecks, trends, and unused resources
- Optimized use of buffer pools (size, object placement)
- Non-disruptive simulation of buffer pool behaviour before making changes
- Long-term analysis of performance data for prediction of future performance and of future resource needs
- Obtain better Hit Ratios to reduce I/O
- Optimize DB2's usage of memory resources

Buffer Pool Analyzer



IBM Software Group

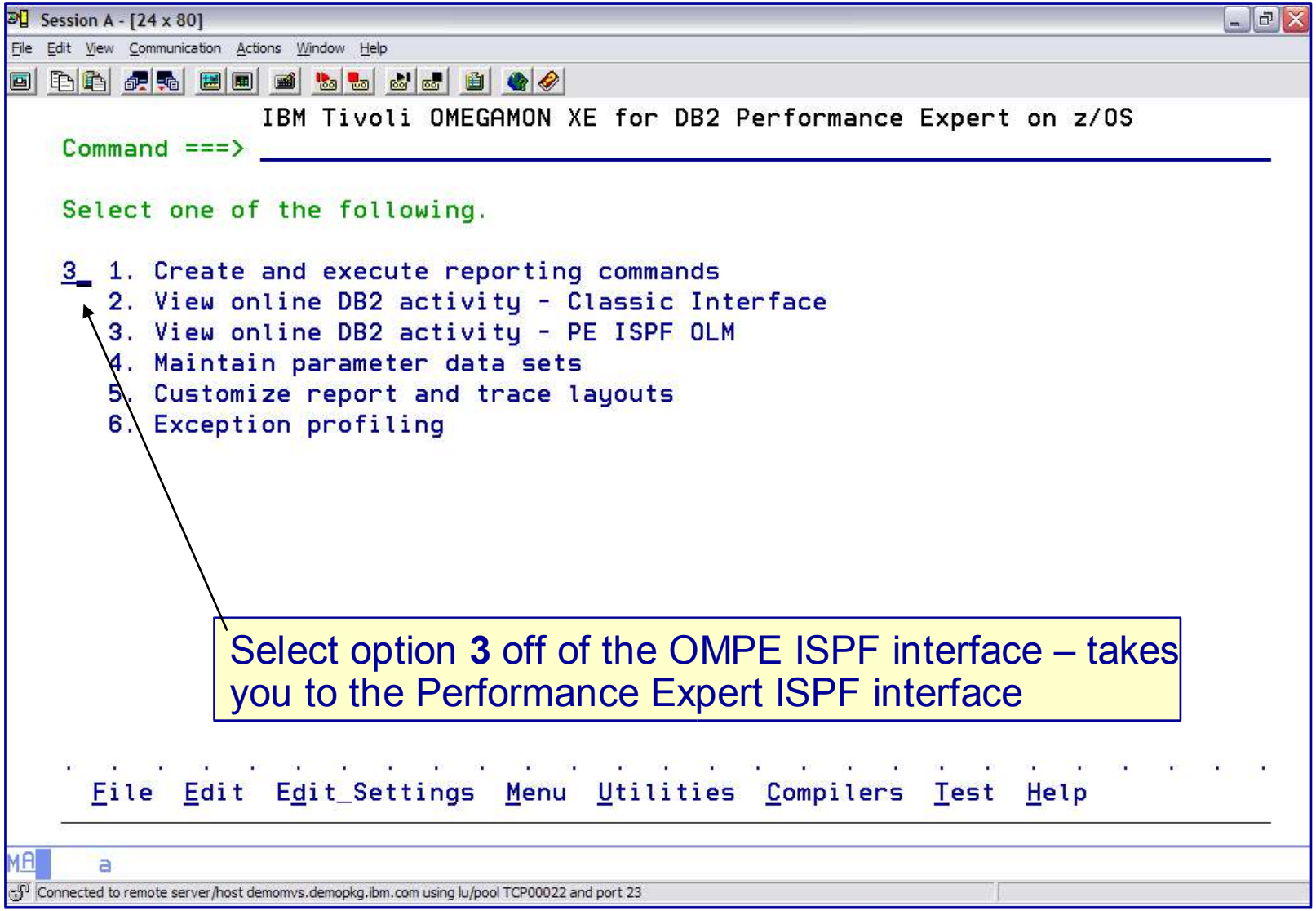
Getting Started with Buffer Pool Analyzer

STEP 1: Trace Data Collection

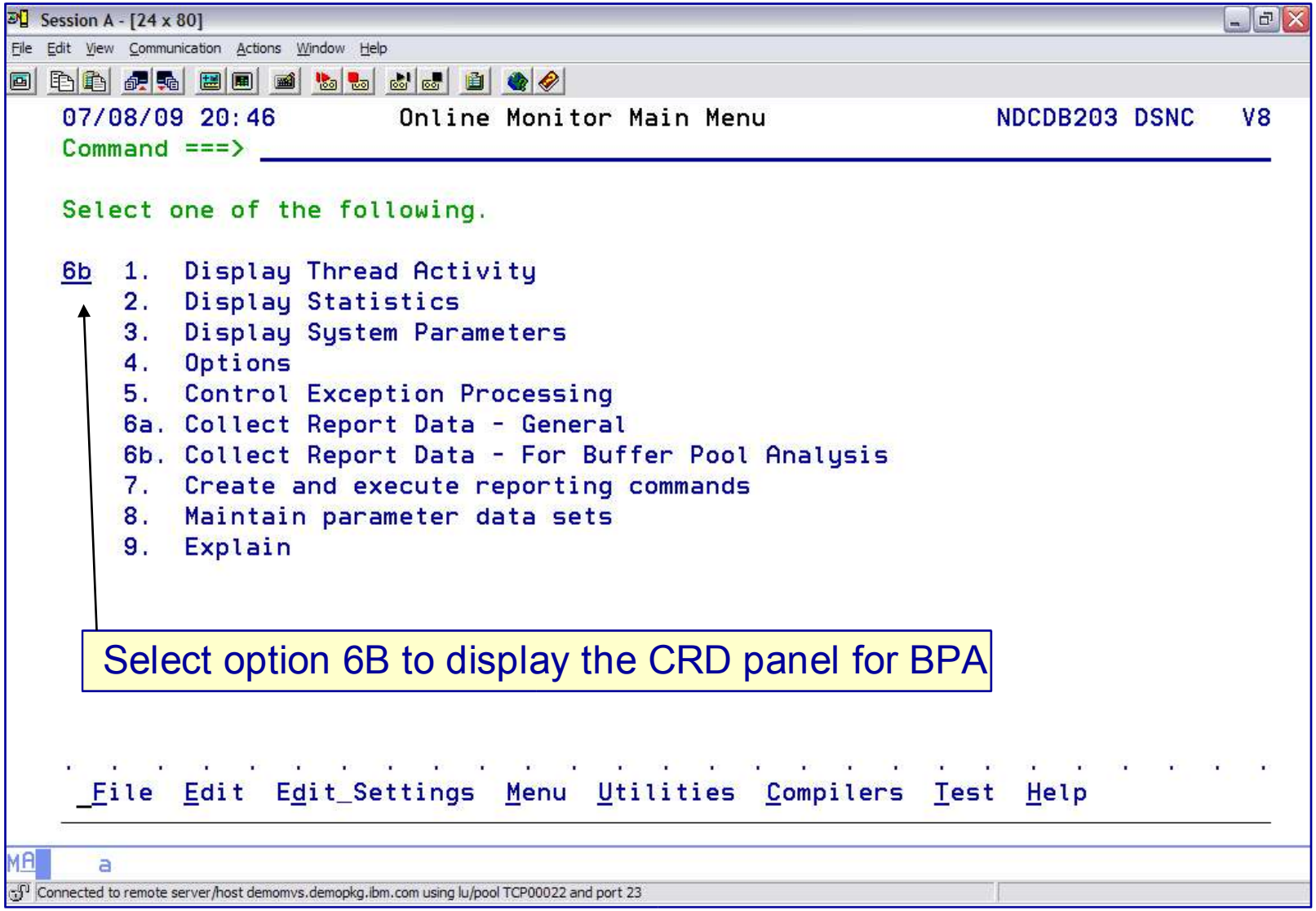
Data Collection - Trace Management

- Can be executed online (ISPF) or Batch
- DB2 Buffer Pool Analyzer uses its own data **collection** to avoid costly overhead
- Continuous collection of **summary data** with minimal overhead
  - Buffer pool statistics and I/O activities for objects
- Collects samples of **detail data** for short intervals
  - Collects all buffer pools events, including I/O and page counts, page references, page updates
- **Detail data** sampling mode under user control
  - for example, trace data every 30 minutes for 30 seconds
- DB2 trace data is stored in a **TSO data set** for further processing

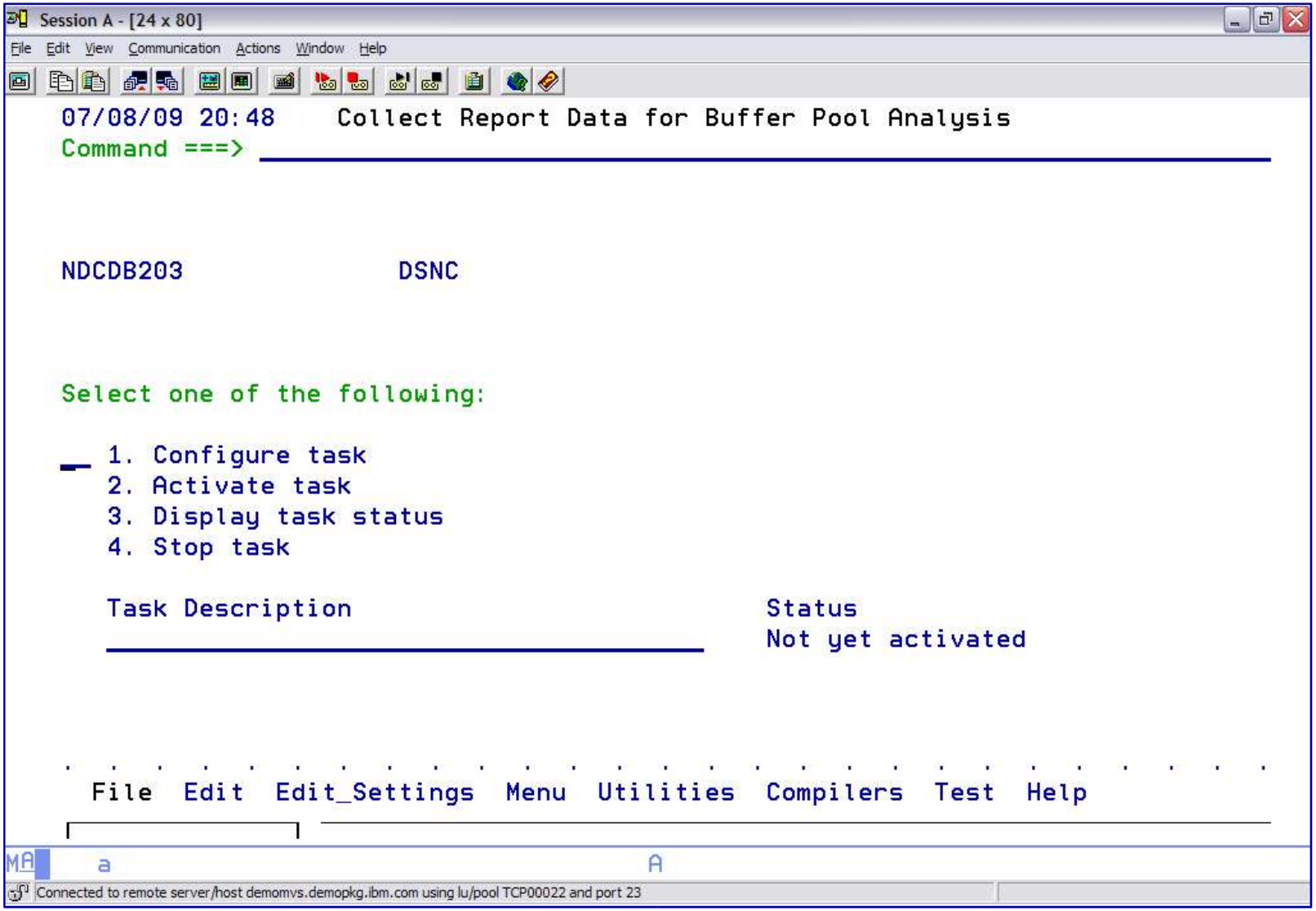
Using Collect Report Data (CRD)



Using Collect Report Data (CRD)



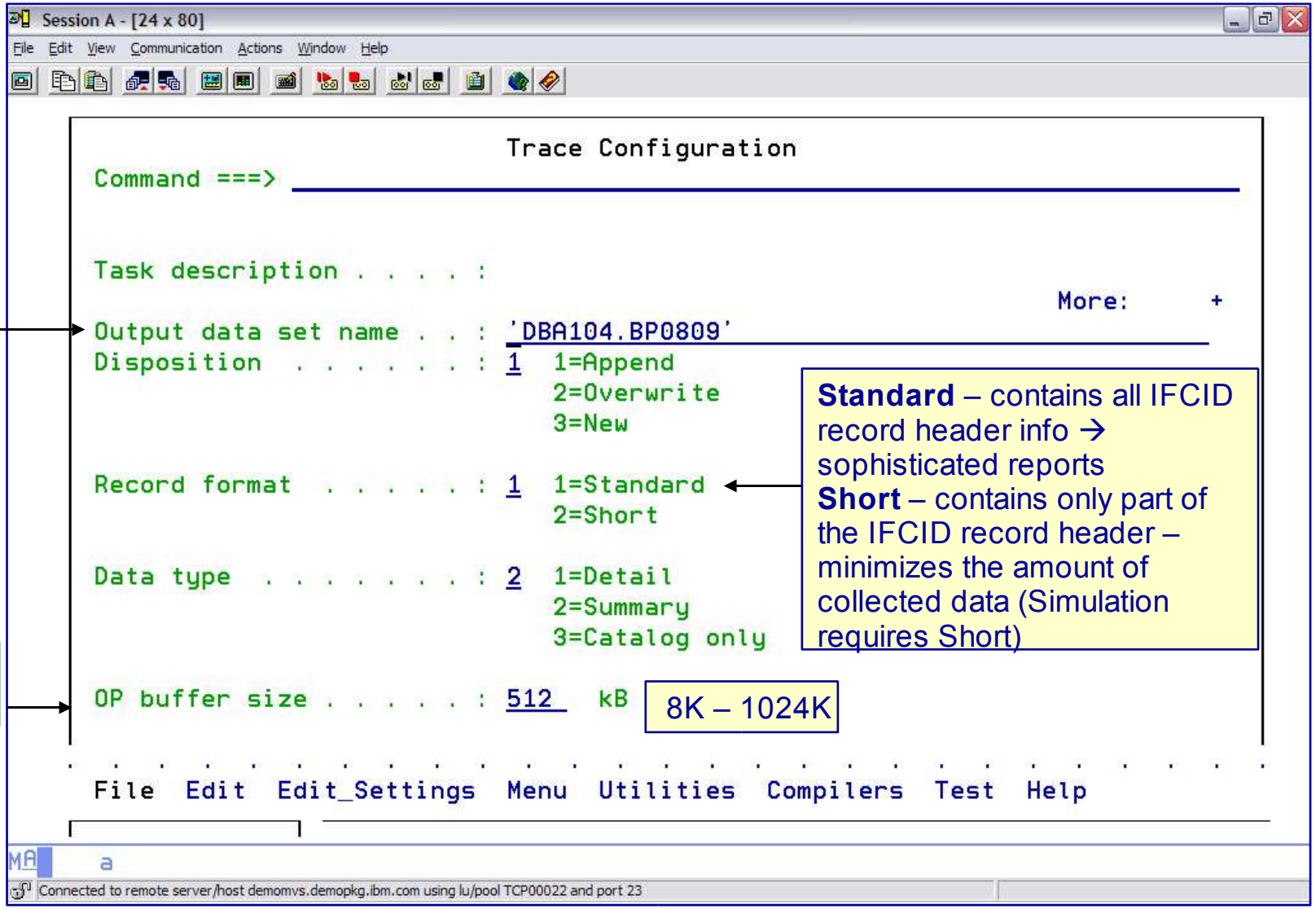
CRD Tasks


Session A - [24 x 80]
File Edit View Communication Actions Window Help
07/08/09 20:48 Collect Report Data for Buffer Pool Analysis
Command ===>
NDCDB203 DSNC
Select one of the following:
1. Configure task
2. Activate task
3. Display task status
4. Stop task
Task Description
Status
Not yet activated
File Edit Edit_Settings Menu Utilities Compilers Test Help
MA a A
Connected to remote server/host demomvs.demopkg.ibm.com using lu/pool TCP00022 and port 23

CRD TASKS - Configure the Trace

Name of the trace output data set

Used to pass trace data to BPA



Standard – contains all IFCID record header info → sophisticated reports
Short – contains only part of the IFCID record header – minimizes the amount of collected data (Simulation requires Short)

Configure the Trace Recommendations

- Output DS Name
  - Should be VB
  - RECL of at least 6000
- Dataset name should have a low level qualifier of TRACE
  - Simulation trace data must have an extension of .TRACE - using the name on both sides eases the download procedure
  - Avoid BPD as a low level qualifier - BPD files must have a file name extension of BPD on the client

Data Collection - Record Format

- **Standard**
  - Includes the complete IFCID record header
  - Select this option if you want to create comprehensive activity reports
  - Standard provides the information that is used by the INCLUDE , EXCLUDE , ORDER ,and SORT options of the BPACTIVITY REPORT command
  - **Do not use** this option if you want to use the data for **simulation**
- **Short**
  - Includes only part of the IFCID record header
  - Select this option if you want to use the data for the workstation-based functions
  - Mandatory for **simulation**
  - For most activity reports Short is also sufficient

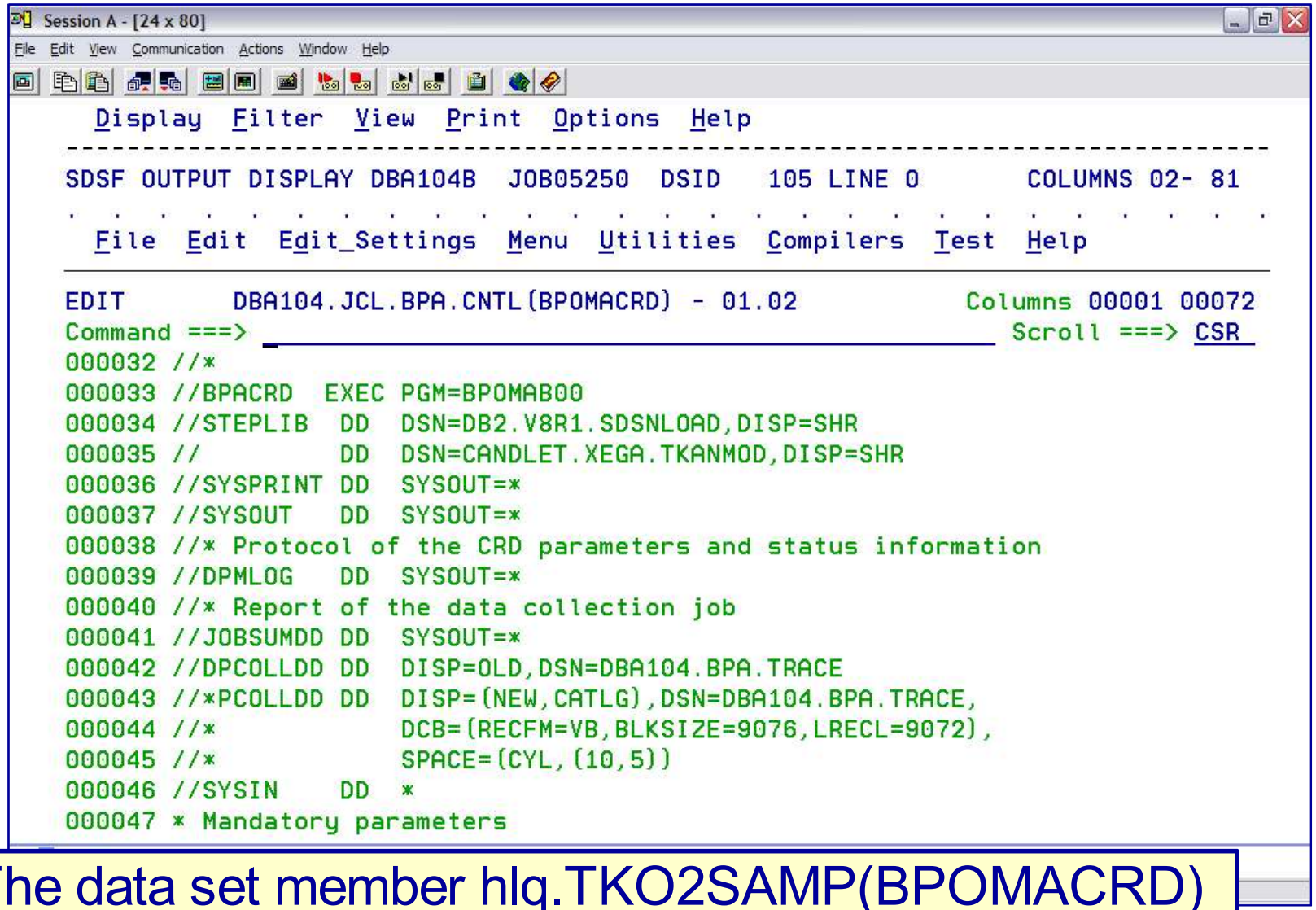
Data Collection - Data Type

- **Detail**
  - Collects buffer pool statistics, catalog data, and buffer pool activity data.
  - Select this option to create Detail Activity reports, bpd files or trace data files for use on the workstation
  - This option is **mandatory for simulation**
- **Summary**
  - Collects buffer pool statistics and catalog data
  - Select this option if you want to create Summary Activity reports
- **Catalog only**
  - Collects only catalog data
  - If the trace data from other sources misses catalog information, you can use the Catalog only option of the CRD function (or the corresponding parameter in a batch job) to collect only catalog information
  - This data is used to enhance trace data that is collected through SMF or GTF

Using Batch JCL to Gather Performance Data

- More suited to regular scheduling via job scheduler than CRD
- For simulation run record format must be SHORT
- To produce only summary reports DATATYPE should be SUMMARY
- To produce detail reports, or to create bpd files or trace data files on the workstation GUI, DATATYPE should be DETAIL
- As well as (or instead of) maximum job duration you can specify a maximum number of records to collect
- Tracing can also be done is sampling mode, as in `SAMPLING (mmm,sss)`, where 'mmm' denotes the interval between two collection periods in minutes, and 'sss' denotes the time in seconds when tracing is active
- Sample JCL can be found in hlq.TKO2SAMP(BPOMACRD)

Using Batch JCL to Gather Performance Data



The data set member hlq.TKO2SAMP(BPOMACRD) contains a typical JCL sample with comments.

Data Collection - Batch

- **Mandatory parameters**

  DB2SSID (DSNB) * DB2 SUBSYSTEM ID

  PLANNAME (BPOPLAN) * DB2 BPA planname

- **Optional parameters**

  RECORD_FORMAT (SHORT) * STANDARD OR SHORT

  DATATYPE (DETAIL) * DETAIL(DEFAULT), SUMMARY, OR CATALOG

  STARTTIME (IM) * IMMEDIATELY OR HH:MM:SS

- **Instead of specifying a start time you can use your batch scheduling**

  DURATION (200S) * MAXIMUM JOB DURATION

  * Time units and u = s for seconds, m for minutes, or h for hours

  * Default is 30m.

  MAX_RECORDS (20000) * MAXIMUM NUMBER RECS

  * collected, optionally in K (=1000) or M (=1000000)

  An example is 25000 which is the same as 25K

Data Collection - Batch

- BUFSIZE (1024)
  - * Specifies the OP buffer size in the DB2 Start Trace command
  - * nnnn indicates the number of KB and ranges from 8 to 1024
  - * Default is 512 for 512 KB

* The following may be activated by removing the first column

- SAMPLING (mmm,sss)
  - * Indicates that tracing is done in sampling mode
  - * mmm denotes
    - * time interval between 2 collection periods
      - * in minutes
    - * sss denotes the time in seconds when DB2 trace is active during a sampling interval
  - * If SAMPLING is omitted (default), DB2 trace data are collected continuously

Data Collection - Record Format

- Ensure that your output data sets are large enough
  - The amount of data that is being collected depends largely on the activity in the buffer pools
- If you are going to collect data for **optimizing the object placements**, ensure that the **DB2 catalog statistics are up to date**
- If you **collect for simulation**, ensure that you collect detail data, in **short format**, for about **20 minutes continuously**
- **Avoid collecting more than 2 GB of data**
  - Simulation function on the workstation can ONLY handle trace data files of up to 2 GB
- If you are going to collect data for **object placement and simulation**, ensure that all requirements described above in item 2 and item 3 are met
  - Furthermore, it is essential that you keep the trace data file and the bpd file together
- Use consistent naming standards on downloaded workstation files

Data Collection - Duration and Stop Conditions

- **Trace Duration**
  - **Continuously**
    - Runs the DB2 trace for the entire tracing period
    - Select this option if you want to use the trace data for simulation
  - **Every x minutes for y seconds**
    - Runs the DB2 trace every x minutes for y seconds, where **x** denotes the specified minutes, and **y** denotes the specified seconds
- **Stop conditions** (either one or both):
  - Elapsed time - specify the number of seconds the trace should run
  - Number of records collected - specify a maximum number of records to be collected
    - Example, 10,000 ,or 100K (for 100 000), or 1M (for 1,000,000).
  - **Simulation** - ensure that you collect trace data for about 20 minutes, respectively 1,200 seconds
    - In the field Elapsed time specify an appropriate value
    - Either deactivate the number of records collected condition, or set it to 13,000,000 (about 2 GB of data), to ensure that the trace does not stop earlier

IBM Software Group

Getting Started with Buffer Pool Analyzer

STEP 2: Creating Activity Reports and BPD files

ON DEMAND BUSINESS

Buffer Pool Analyzer Reports

- Provides detailed statistical reports about
  - Buffer pool usage
  - Object usage
- The report can be tailored to generate various views:
  - Ranking information by different identifiers (for example, buffer pool, plan, object, PRIMAUTH)
  - Sorting the data by get pages, sequential prefetch, or synchronous reads
  - Filtering information according to buffer pool, or plan
- The report presents the most important objects
- Use FILE for subsequent loading into DB2 tables for additional analysis by SQL queries
  - New run time improvement options using FILE LEVEL (SUMMARY) and / or FILE ACTIVEOBJECTS

REPORT Data Sets

- Sample JCL in hlq.TKO2SAMP(BPOQBTCH)
- Batch job
  - INPUTDD - data set containing the trace data
    - Usually from TRACE file collected by CRD or BPOMACRD
    - Could use data from GTF or SMF data sets
    - Can process multiple input data sets
  - SYSOUT - messages from DFSORT
  - JOBSUMDD - if specified, contains
    - Job summary log
    - IFCID frequency distribution log
    - Format is RECFM=FBA, LRECL=133, BLKSIZE=6251
  - DPMLOG
    - Contains messages from the BPA command processor
    - Format is RECFM=FBA, LRECL=133, BLKSIZE=6251

REPORT Data Sets

- Batch Job
  - BPRPTDD
    - Output of the BPACTIVITY REPORT subcommand
    - Format is RECFM=FBA, LRECL=81, BLKSIZE=8100
    - Multiple REPORT subcommands, output is written to BPRPTDD in sequence
  - BPFILDD1
    - Output from the BPACTIVITY FILE subcommand
    - BPD file that can be used on the client for viewing performance data and optimizing object placement
    - Contents can be loaded into DB2 tables
    - Format is RECFM=VB, LRECL=9072, BLKSIZE=9076
    - Recommend a low level qualifier of BPD

REPORT Data Sets

- Batch Job
  - BPWORK
    - Optional
    - Determines where BPA stores temporary data
    - Do NOT specify DUMMY or DISP=MOD
  - SYSIN
    - Mandatory
    - Commands to run the job stream

REPORT Commands

BPACTIVITY

REPORT

FROM / TO

INCLUDE / EXCLUDE

LEVEL DETAIL / SUMMARY

DDNAME BPRPTDD

ORDER

BPACTIVITY

FILE

FROM / TO

INCLUDE / EXCLUDE

LEVEL DETAIL / SUMMARY

DDNAME BPFILDD1

ACTIVEOBJECTS

Note: FILE subcommand can only be used 1 time in a batch job

If FILE and REPORT are used in the same batch job with the LEVEL subcommand, the same LEVEL is required for both

REPORT Commands

- **FROM / TO**
  - Use to selectively use trace data from input data sets for activity reports or bpd files
  - Start date (mm/dd/yy) and time (hh/mm/ss.th)
  - End date and time
  - Use to
    - Isolate a peak load period from the collected data
    - Segregate multiple periods from a single input data set - create several BPD files
    - Do not use if doing long term analysis

REPORT Commands

- **INCLUDE / EXCLUDE**
  - Use to selectively use trace data from input data sets for activity reports or bpd files
  - Limited use with the FILE subcommand - recommendation is to avoid using them with FILE
    - If used with Object Placement, the data would not be reliable

REPORTS - INCLUDE Clause

IDENTIFIER	MEANING	AFFECT SUMMARIES	NO EFFECT W/ SHORT REC FORMAT
BPID	Buffer pool ID	Yes	
CONNTYPE	Connection type		Yes
ENDUSER	End user name		Yes
PLANNAME	Plan name		
PRIMAUTH	Primary authid		
PSTYPE	Pageset type (table or index)		
QPAGESET	Combination of DBASE and page set	Yes	
TRANSACT	End user TXN name		Yes
WSNAME	End user Workstation ID		Yes

- ✓If the input data was collected with a **short record format**, some record header information is not present in the trace records.
- ✓If you specify an identifier for which no header information is available, it will have no effect in reports or bpd files.

REPORT Commands

- **LEVEL**
  - REPORT
    - Use to create a SUMMARY Report
    - Default is DETAIL Report
  - FILE
    - LEVEL SUMMARY - reduces processing time and the size of the BPD file
- **DDNAME**
  - Use to specify a DDNAME other than the default for output reports or bpd files

REPORT Commands

- **ORDER**
  - Use with REPORT to influence the aggregation level and sequence of reported statistics
  - Default - reports are aggregated by BPID and a combination of database and page set (QPAGESET)
  - TOP - topmost entries
    - Skips reporting of objects having low usage rate
  - **SORTBY** - modify the sort sequence

ORDER (BP10-CONNTYPE
SORTBY (CONNTYPE,GETPAGE)

Aggregated by BPID and CONTYPE
Sorted by CONNTYPE and # GETPAGES

REPORT - ORDER Clause

IDENTIFIER	MEANING	No effect w/ Short Record Format
BPID	Buffer pool id	
CONNTYPE	Connection type	Yes
ENDUSER	Enduser name	Yes
PLANNAME	Plan name	
PRIMAUTH	Primary authid	
PSTYPE	Pageset type (T - table, I - index)	
QPAGESET	Combination of DBASE and pageset	
TRANSACT	End user transaction name	Yes
WSNAME	Workstation ID	Yes

- ✓If the input data was collected with a short record format, some record header information is not present in the trace records
- ✓If you specify an identifier for which no header information is available, it will have no effect in reports or bpd files

REPORT - SORT Clause

SORT FIELDS	DESCRIPTION	VALID FOR DETAIL REPORT	VALID FOR SUMMARY REPORT
GETPAGE	Total Getpages	Yes	
READSEQ	Total Seq. reads	Yes	
READREQ	Total Reads	Yes	
READSYNC	Total Synch Reads	Yes	
MISRAND	GP - Miss random	Yes	
MISSASYN	GP - Miss synch	Yes	
READPAGE	Total Read page	Yes	
WRITEPAGE	Total Write page	Yes	
WRITEREQ	Total writes		Yes
ASYNCH READ	Total asynchronous reads		Yes
ASYNCH WRITE	Total asynch writes		Yes

REPORT Commands

- **ACTIVEOBJECTS**
  - Only used with the FILE subcommand
  - Use to only get information about active objects
    - Those with BP activity
  - Reduces size of BPD file

REPORT - Accuracy

- SUMMARY REPORT
  - Use BP and data set stats data from DB2
  - BP behavior on an overall basis
  - Statistical data is recorded by intervals
  - Time frame does not always match the start / end of the interval
  - Partial intervals may affect accuracy
  - 2 major sections
    - Statistics / BP
    - I/O statistics on the data sets backing the BP's
      - ORDER, SORTBY, TOP
- DETAIL REPORT
  - How applications access the BP
  - Describes how requests are processed that are issued by the data manager component
  - Uses activity data from DB2 - event based
  - Collect precise information about every single activity during the specified time frame

REPORT - Sample Statements

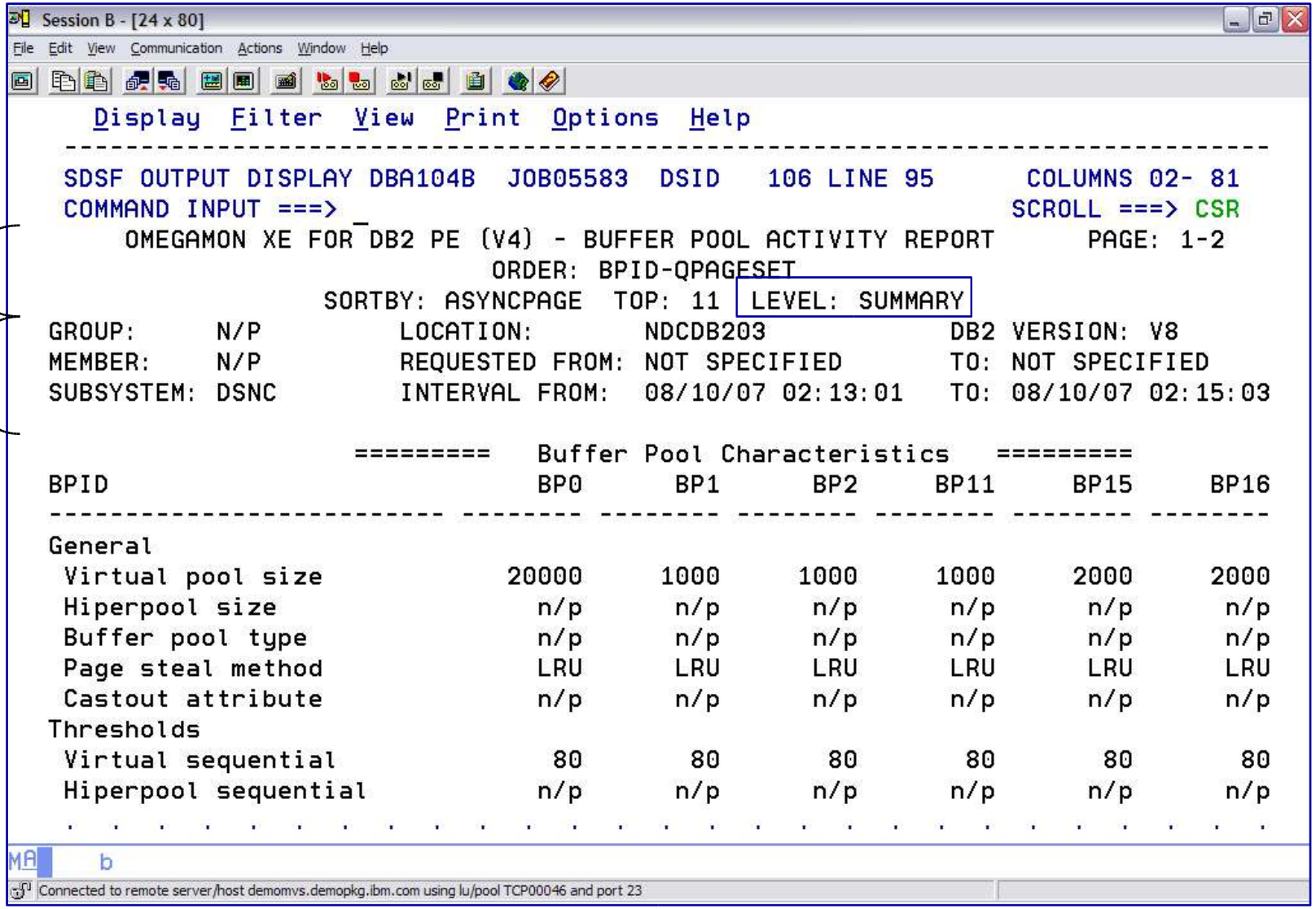
Set GMT

```
GLOBAL
   timezone (-1:00)
  BPACTIVITY
  REPORT
      LEVEL(DETAIL)
      ORDER(PLANNAME-BPID-QPAGESET
          SORTBY(PLANNAME,GETPAGE) TOP(5))
 *    Output goes to DD name BPAREP2
      DDNAME(BPAREP2)

* * *                                                  * * *
* * *  Up to 5 Report Clauses per Step Execution       * * *
* * *                                                  * * *


 EXEC
```

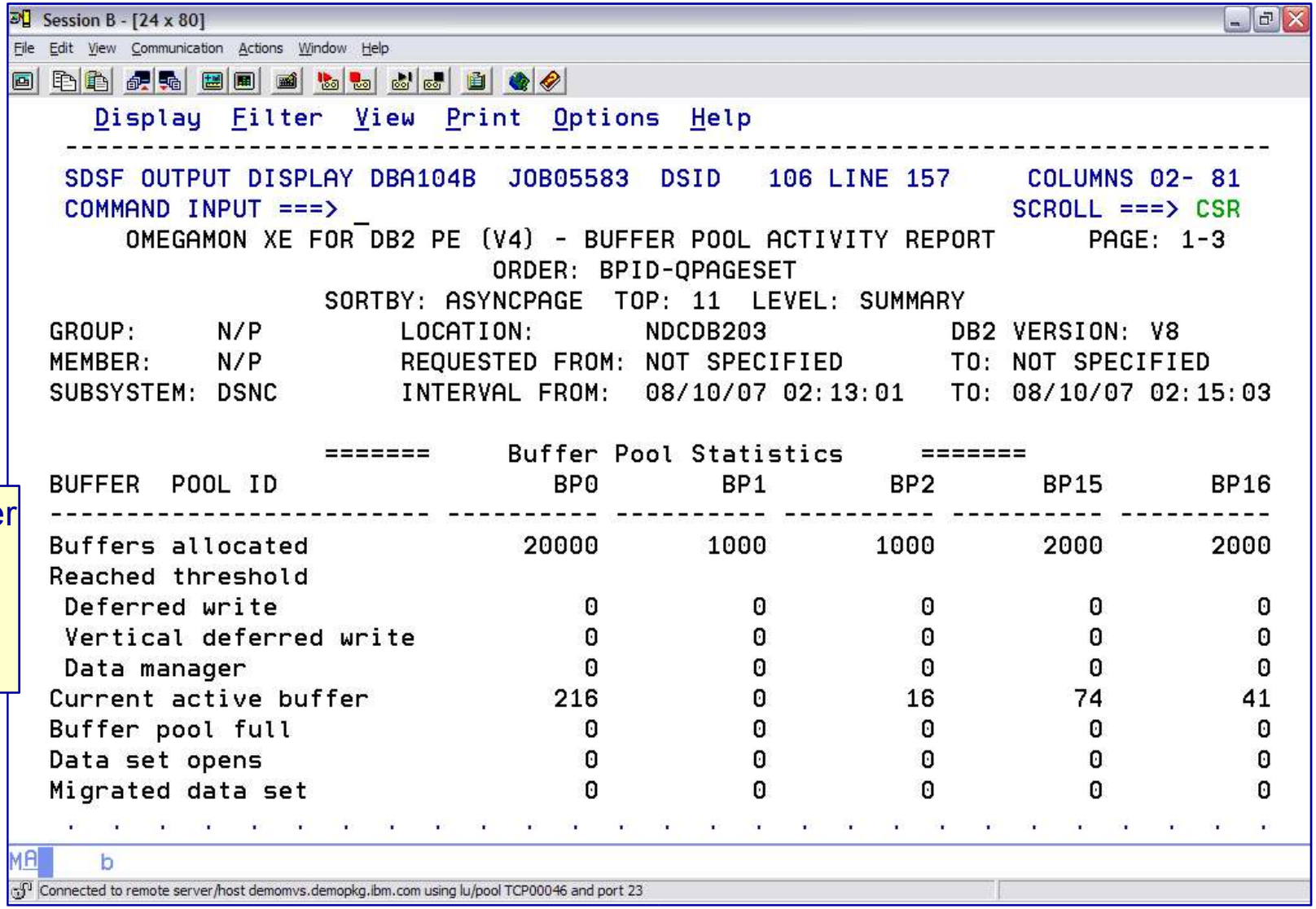
Report Output
Summary

Header – top of every report

Statistics section

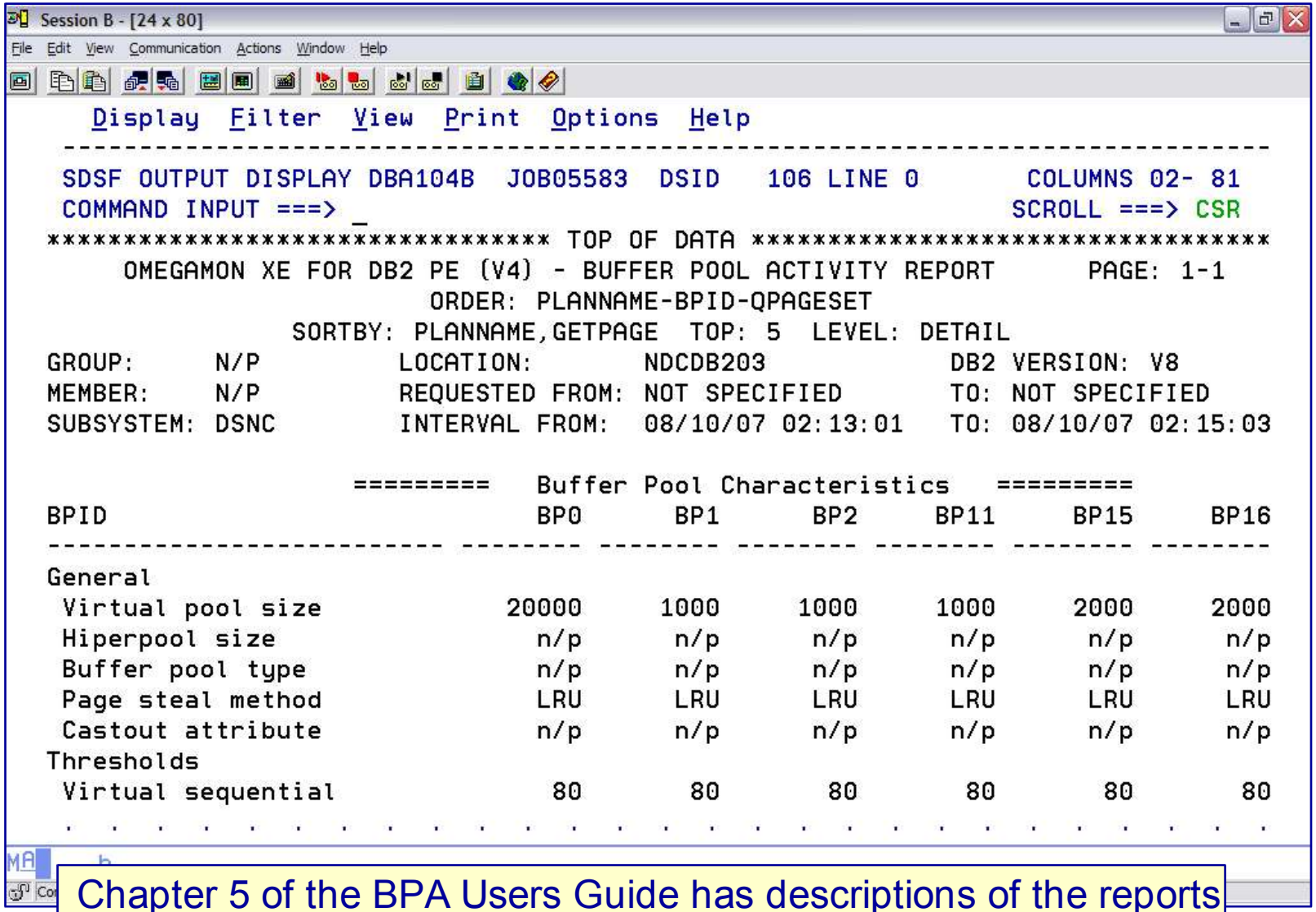


Report Output
Summary



If there is an * after a number, indicates that the counter should typically be 0

Report Output
Detail

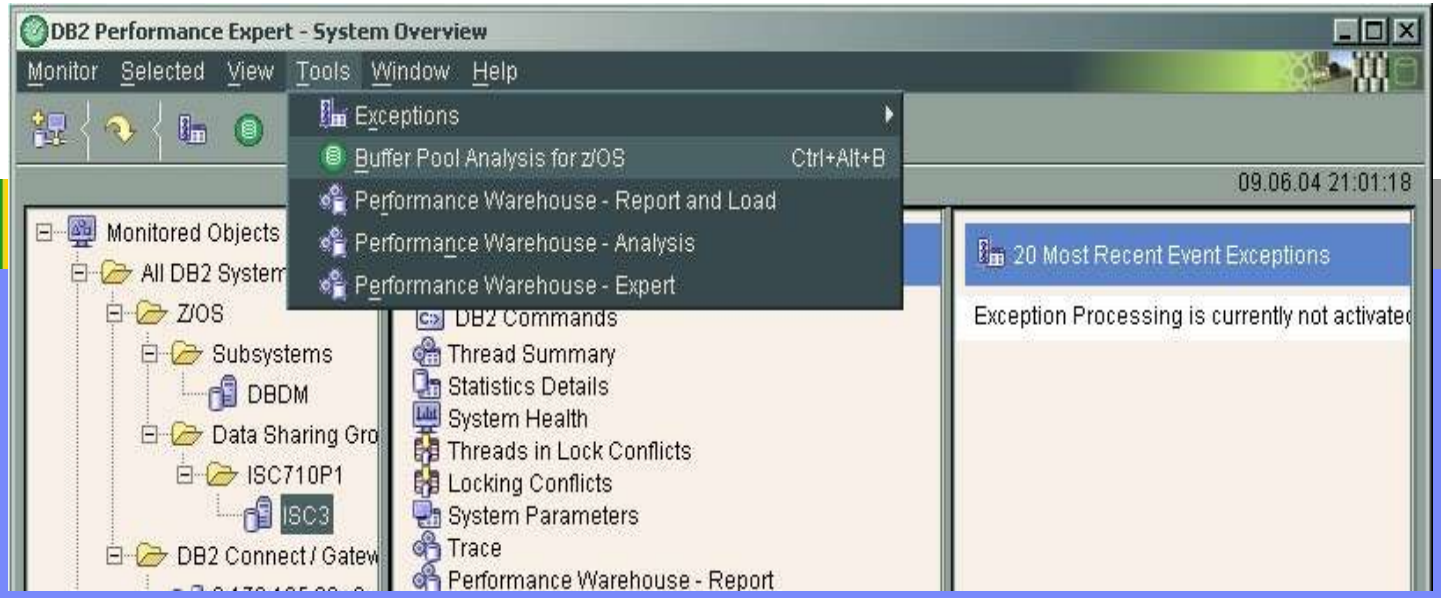


Chapter 5 of the BPA Users Guide has descriptions of the reports

IBM Software Group

Getting Started with Buffer Pool Analyzer

GRAPHICAL REPORTS



ON DEMAND BUSINESS

Work Station Requirements

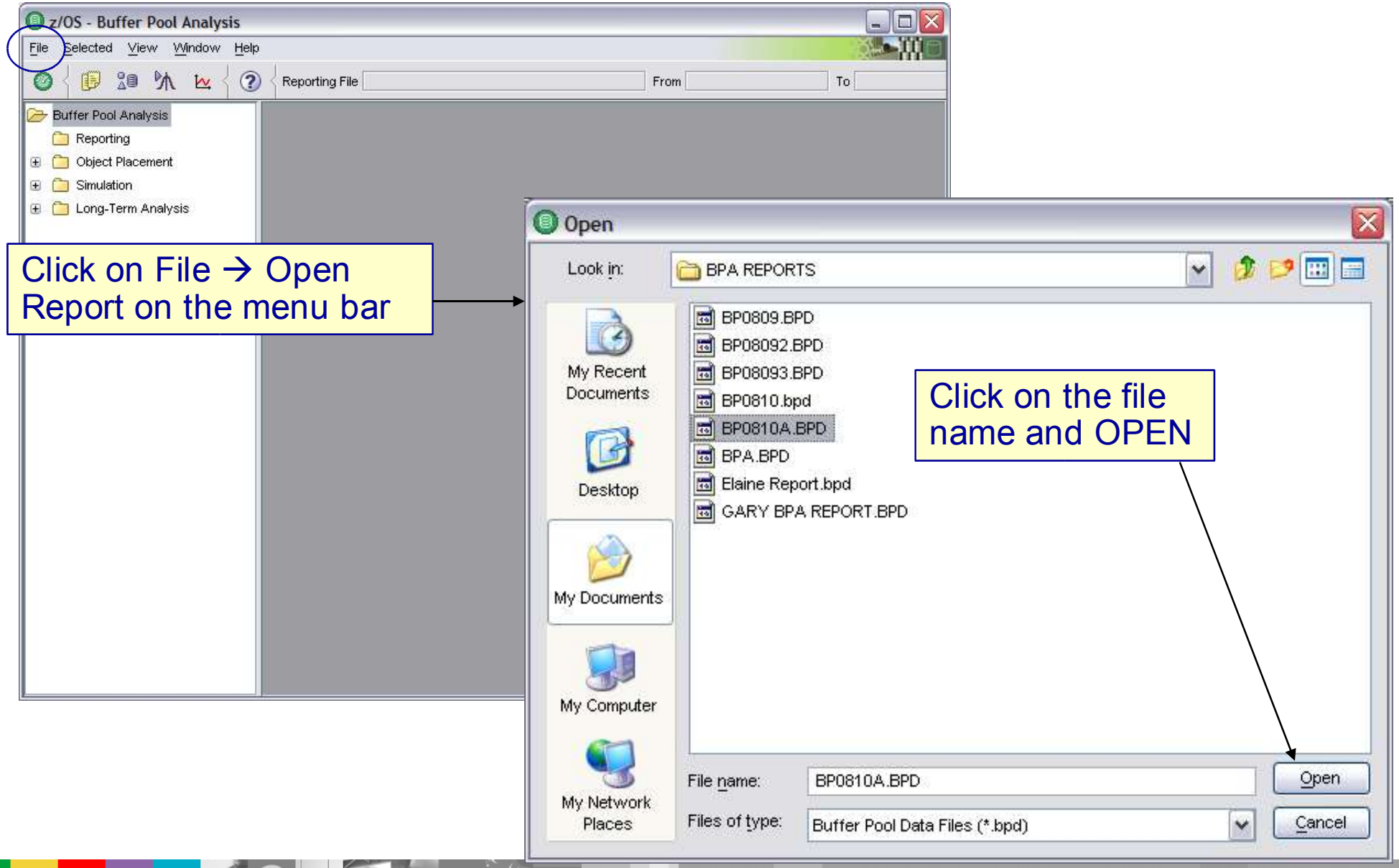
General Comments

- Workstation should have at least **40 MB RAM**
  - Check this in the Windows Task Manager
  - Close other applications, if you receive an Out of memory message
  - Note that free memory requirements increase with the number of objects to be treated
  - Memory requirements increase as the number of objects increase
    - 1 000 objects require about 60 MB
    - 25 000 objects require about 90 MB
    - 100 000 objects require about 230 MB
    - 200 000 objects require more than 500 MB
- The data used for this function reflects the performance for the interval for which trace data was collected and for the time frame that was specified with the GLOBAL and the BPACTIVITY command when the bpd file was created
- Use either sampling or capture multiple BPD files for a representative sampling of data

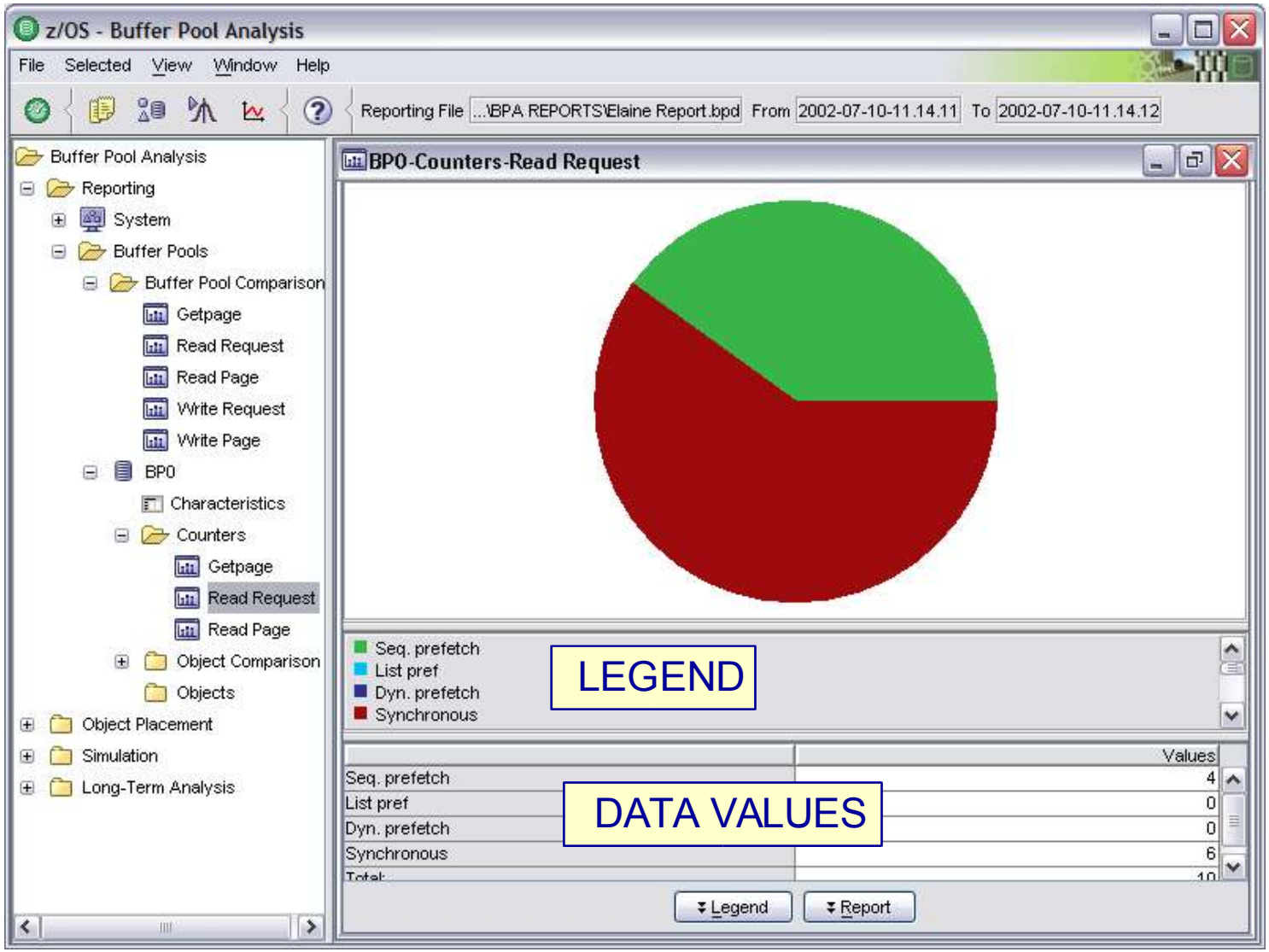
Send Report and Trace Files to the Workstation

- To be able to perform buffer pool analysis and object placement you need to **create .bpd files and transfer these to the workstation**
- For **simulation** runs you need the **trace file created by Collect Report Data**
- Include a **FILE subcommand** in the Buffer Pool Analyzer command stream in the z/OS batch job to create a **.bpd file**
  - The FILE subcommand can specify INCLUDE, EXCLUDE, FROM - TO and DDNAME options to specify the data to be written to the bpd file and the DD name of the output file
- Once the files have been created, use **FTP** or your 3270 emulator program to copy the .bpd and/or trace files into a directory onto the workstation where the PE Client is installed
  - Use the **binary option** when transferring the data
- To start Buffer Pool Analyzer from the workstation GUI, select 'Buffer Pool Analysis for z/OS' from the Tools menu bar item

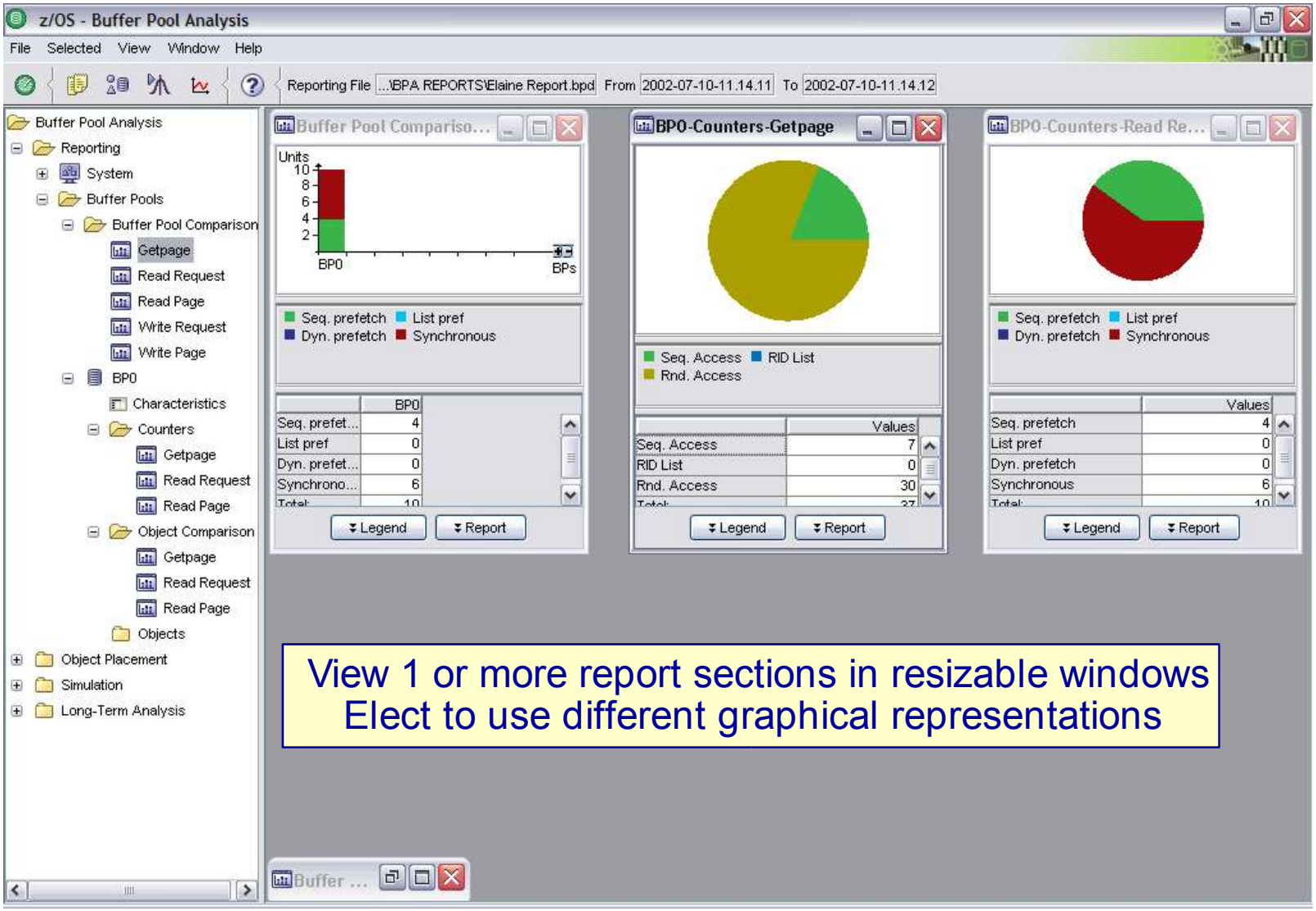
Open a .bpd (Report) File in BPA GUI



Viewing Report BPA Sections



Viewing Report BPA Sections



IBM Software Group

Getting Started with Buffer Pool Analyzer

OBJECT PLACEMENT

ON DEMAND BUSINESS

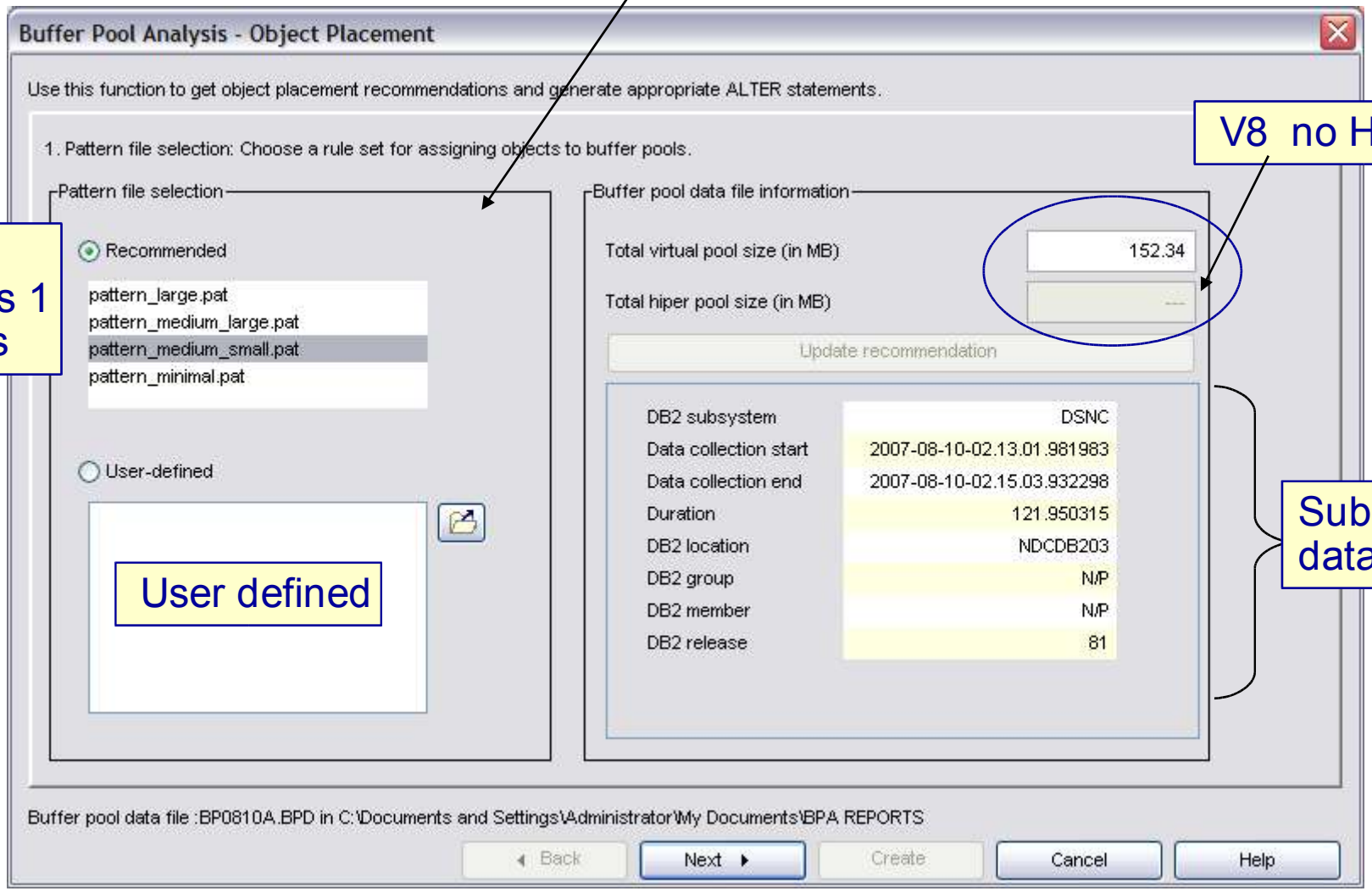
Object Placement

- ☑ Collect trace data
  - CRD
  - Batch job - hlq.TKO2SAMP(BPOMACRD)
- ☑ Create bpd file
  - Batch job - hlq.TKO2SAMP(BPOQBTCH)
- ☑ FTP bpd file to client
  - BIN
  - .bpd extension
- ❑ Bring up the GUI interface
  - ❑ Click on TOOLS and select Buffer Pool Analyzer
  - ❑ Click on FILE and select Object Placement
  - ❑ Select a .bpd file

Object Placement

Select a Pattern File

Select a recommended (supplied) or user-define pattern file.



BPA recommends 1 of 4 patterns

V8 no Hiperpool

User defined

Subsystem data

Select a different pattern file and / or adjust the available memory for BP's

Object Placement

Pattern Files

- **List of rules** that determine which objects (table spaces and indexes) should be placed in which buffer pool according to each object's characteristics
- Initially, Buffer Pool Analyzer **recommends one of four pattern** files based on the total buffer pool size (virtual pool and hiperpool) that is determined from the content of the bpd file
- The **rules** in each pattern file are **predefined**
- If you edit a pattern file and save it under a user-defined pattern, Buffer Pool Analyzer recommends this user-defined pattern file whenever a bpd file from the same DB2 subsystem is processed by the object placement wizard

Object Placement
Pattern Files

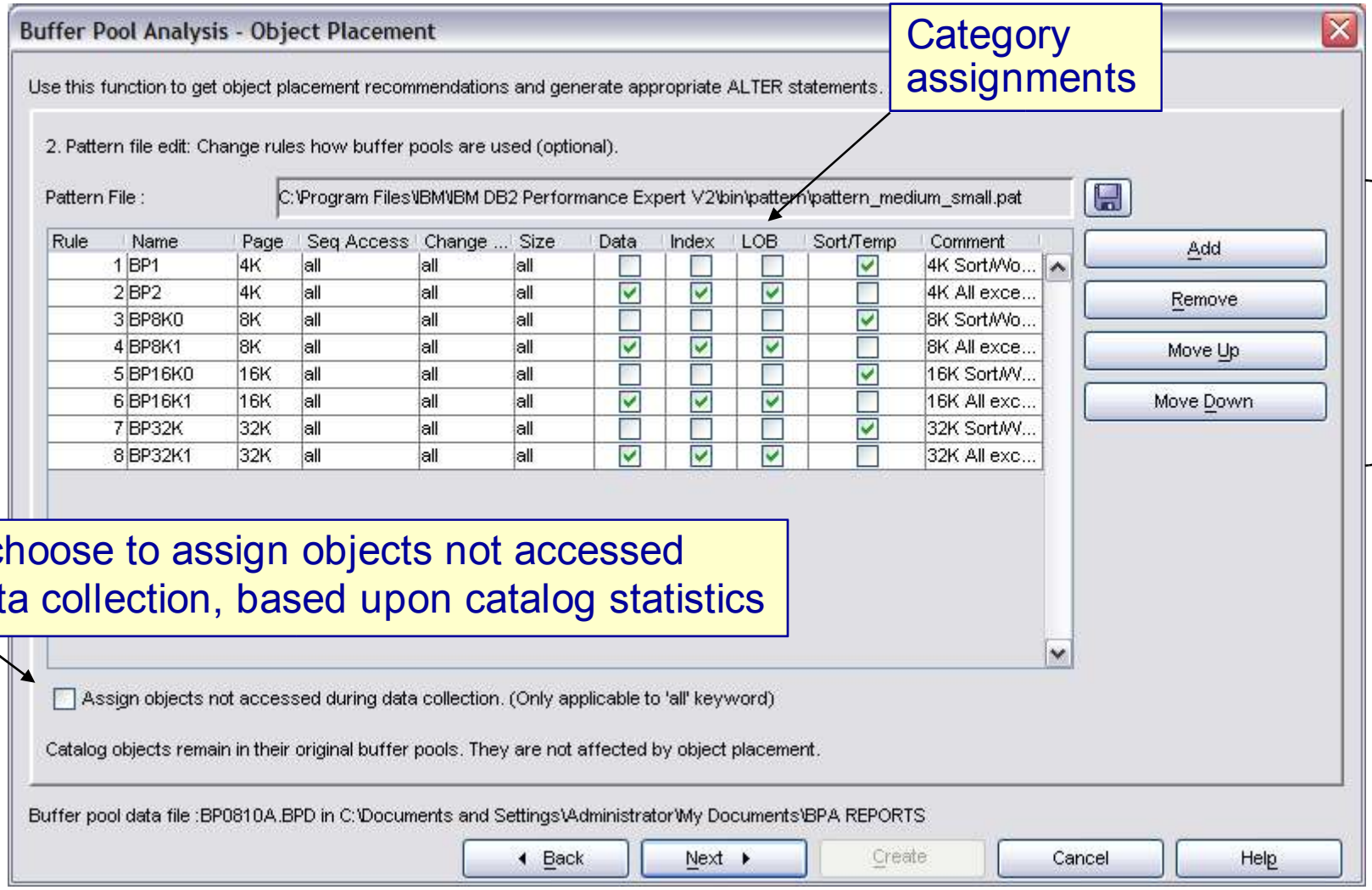
- The Total Virtual Pool size and Total Hiperpool size values are determined from the content of the bpd file and reflect the values at the time the data was collected from the DB2 subsystem
  - This can be modified to simulate a placement analysis with additional memory allocations
- Object placement rules are stored in the selected pattern file.
- The list of rules acts like a series of **filters**
  - The object placement wizard processes all objects (table spaces and indexes) through these rules, starting at the top of the list
  - Once the characteristics of an object match the criteria for a specific buffer pool, the object is assigned to this buffer pool

Object Placement
Pattern Files

- Each rule specifies a series of object characteristics as criteria for a buffer pool
  - If an object matches **all** criteria of a rule, the wizard recommends it for placement in the corresponding buffer pool
  - If an object does not match all criteria of a rule, it is passed to the next rule for evaluation
- An object matches a rule if
  - Page Size, Seq Access, Change Rate and Size match
  - And if an object is one of the selected data types Data, Index, LOB, or Sort/Temp
- Objects with similar characteristics are assigned to the same buffer pool
- Place rules with restrictive criteria at the top of the list, and those with more general criteria at the bottom
  - Otherwise, the more specific rule might never become active

Object Placement

Edit a Pattern File



Object Placement
Rule Elements

- **Name** - Buffer Pool Name (BP0, BP16K9)
- **Page** - Shows the size of each buffer pool page
  - The size is implicitly extracted from the name of the buffer pool
- **SEQ Access** - Specify a % range as criteria for an object 's sequential accesses
  - Objects with sequential accesses within the specified range, out of all accesses
- **Change Rate** - Specify a % range as criteria for an object 's change rate
  - Objects with a change rate within the specified range, out of all accesses, are assigned to the specific buffer pool
- **Size** - Specify a size range as criteria for an object 's size
  - Objects with a size within the specified range are assigned to the specific buffer pool
- **Data** - Assigns table space objects (if other criteria are met)
- **Index** - Assigns index space objects (if other criteria are met)
- **LOB** - Assigns LOBs
- **Sort/Temp** - Assigns Sort/Temp objects (4k or 32k only)
- **Comment –** Descriptive text for each rule

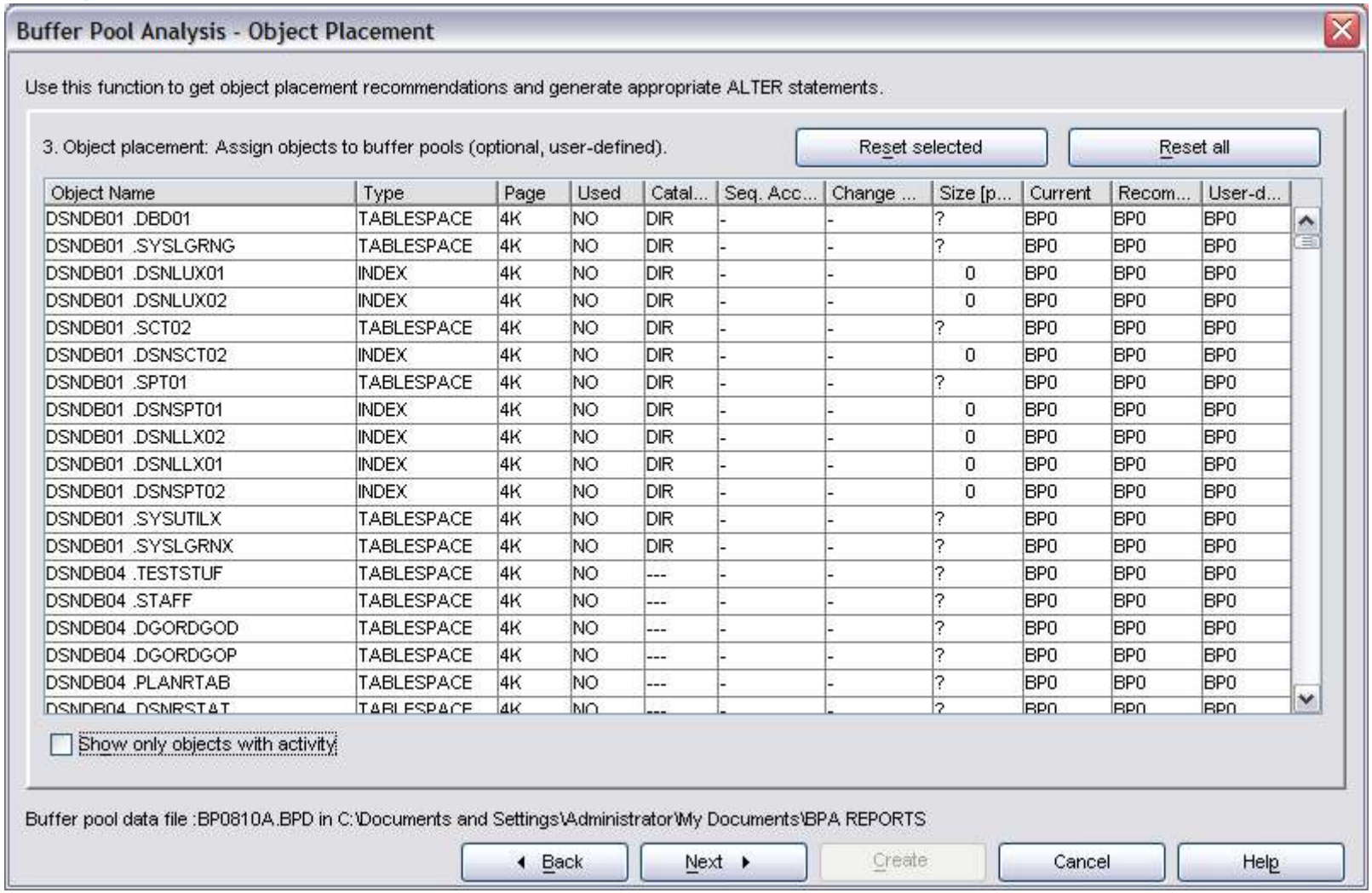
Object Placement
Rule Ranges

- You can use the following notations to express percentage ranges:
  - -50 is the same as 0 to <50%
  - 50-is the same as 50 to =100%
  - ALL is the same as 0 to =100%
- If the *Assign objects not accessed during data collection* check box is selected, ALL also includes objects for which no change rate characteristics could be determined (marked as N/C)
- For example, a percentage of 50-80 considers objects that are changed 50 to less than 80 times out of 100 accesses
- Sizes work in a similar manner:
  - -12 is the same as 0 to <12 pages
  - 50-is the same as 50 to the maximum object size pages
  - All is the same as 0 to the maximum object size pages, and also includes objects for which the size could not be determined

50 – 80% means >= 50 but < 80
80 – 90% means >= 80 but < 90

Object Placement

Assign Objects to BP's



Object Placement
Recommendations

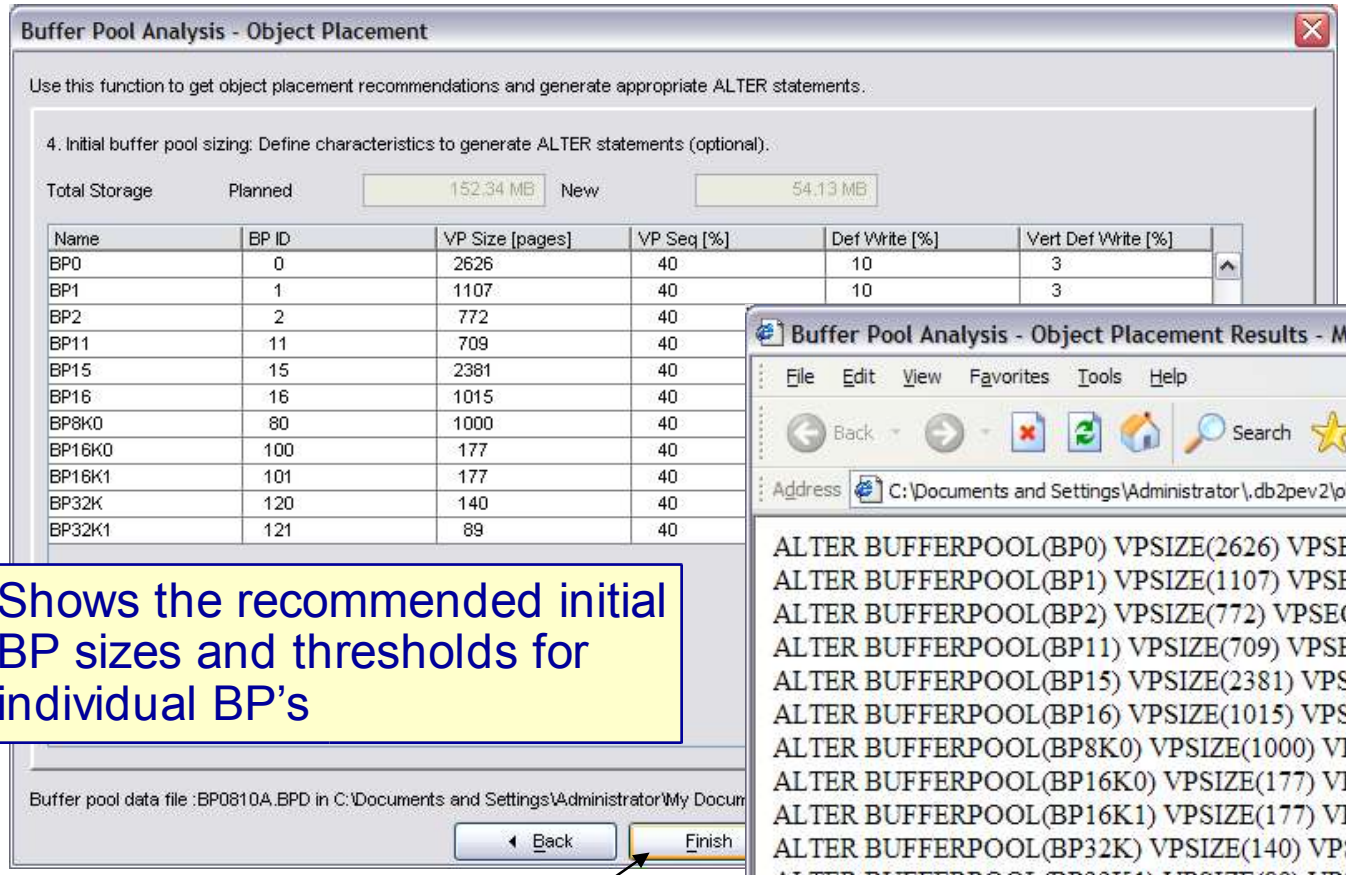
- Buffer Pool Analyzer at this point has scanned the bpd file and has analyzed the characteristics of each table space and index
  - Current column shows the placement of objects at the time the data was collected
  - Recommended column shows the recommended object to buffer pool assignments, as calculated by Buffer Pool Analyzer
    - The calculations are based on the placement rules for each buffer pool and the object's characteristics
  - User-defined column is initially identical with the Recommended column, and is intended to allow you to change assignments
    - You can assign specific objects to buffer pools other than the ones determined by the placement rules

Object Placement
Recommendations

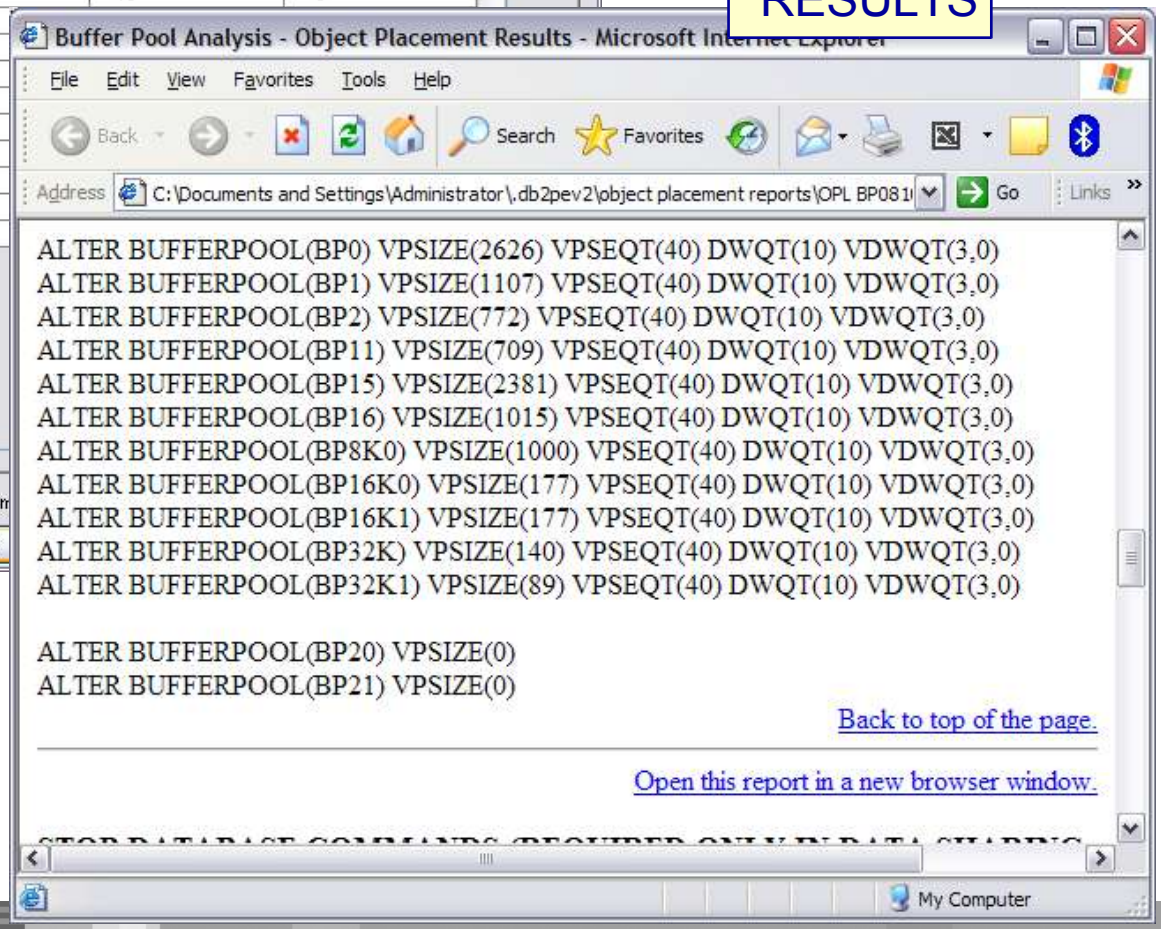
- To change the assignment of an object - double-click the appropriate buffer pool name in the User-defined column and edit the buffer pool name
  - Then press Enter or select a different field
- To reset a User-defined assignment of an object to the Recommended assignment, select the object and click Reset selected
- To select a range of successive objects, click the first object, then, holding the Shift key, click the last object in the sequence
- To select several separate objects, click the first object. Then, holding the Ctrl key, click the other objects as required
- To select all objects, press Ctrl+A
- Note that you can sort the list by clicking a column header of choice one or more times - small arrows indicate the sort order

Object Placement

Setting the Initial BP Sizes and Characteristics



RESULTS



Shows the recommended initial BP sizes and thresholds for individual BP's

Click on FINISH

Object Placement
Applying Changes

- Run the ALTER statements through SPUFI
- If BP size set to 0, double check to see if this BP is no longer used
  - May not have shown any activity for the time of the stats
- If possible, stop the DB before applying changes (especially in a data sharing environment)
- If virtual BP size is limited OR the proposed size is < than the original - apply changes that reduce the sizes of BP first
- After applying ALTER INDEX and ALTER TABLESPACE statements, reassignments to different BP's remain pending until DB2 happens to close and reopen the data sets of the changed page sets

Object Placement
Performance Tips

- Objects w/ similar characteristics should be grouped and placed in the same BP
  - Separation of sequential processing from random provides benefit
- Assign TS and IS into separate BP's that have sequential and random data access
- Do not place any other objects in BP used by the WORK / SORT datasets (DSNDB07)
- BP0 - Exclusive for DB2 Catalog and Directory objects
- Distribute objects in different pools based on
  - Access intensity
  - # buffers in the page set
- Validate the recommended values for DWQT and VDWQT
  - Avoid over-running the I/O subsystem
- FIFO page steal algorithm is recommended (ALTER BP PGSTEAL)
- BP's having high I/O rates, consider the long term page fix option (DB2 V8)
  - Fixes the BP in real storage for an extended period of time
  - Reduces I/O's for I/O intensive BP

Tuning DSNDB07

- Critical part of overall system and bp tuning
- Used and accessed differently than other objects
- Prefetch is only 8 pages
- BPA Sys Hit Ratio may be higher than Application Hit
  - Based on No_Reads - No_Reads occur when new data is created and a page must be created
- Pages written when DMTH and IWTH are hit
  - A common problem is hitting these threshold, especially when there are more than 3 workfiles
- When the entire sort can fit in the pool set the thresholds high

Tuning DSNDB07

- **When Sorts are too large to fit in pool and I/O is always necessary Set threshold extremely low**
  - VDWQT as low as 1%
  - DWQT as low as 5-10%
- **VPSEQT - 95-99% There will be some random I/O**
  - buffers <= 1000 : 95%
  - buffers <= 4000 : 98%
  - buffers <= 8000 : 99%

Page Stealing

- Page stealing algorithm (PGSTEAL). When DB2 removes a page in the buffer pool to make room for a newer page,this action is called stealing the page from the buffer pool
- LRU
  - Default Method that DB2 uses most of the time
  - This means that it takes away pages that are not used so that more recently used pages can remain in the virtual buffer pool
- FIFO
  - With this simple algorithm, DB2 does not keep track of how often a page is referenced - the pages that are oldest are moved out, no matter how frequently they are referenced
  - Use FIFO for buffer pools that have little or no I/O; that is, the table space or index remains in the buffer pool
  - It can reduce internal DB2 latch contention in environments that require very high concurrency

IBM Software Group

Getting Started with Buffer Pool Analyzer

SIMULATION

ON DEMAND BUSINESS

Simulation

- ☑ Collect statistical information
  - CRD
  - Batch job - hlq.TKO2SAMP(BPOMACRD)
- ☑ FTP trace file to client
  - BIN
  - .trace extension
- ❑ Bring up the GUI interface
  - ❑ Click on TOOLS and select Buffer Pool Analyzer
  - ❑ Click on FILE and select Simulation
  - ❑ Select a .trace file

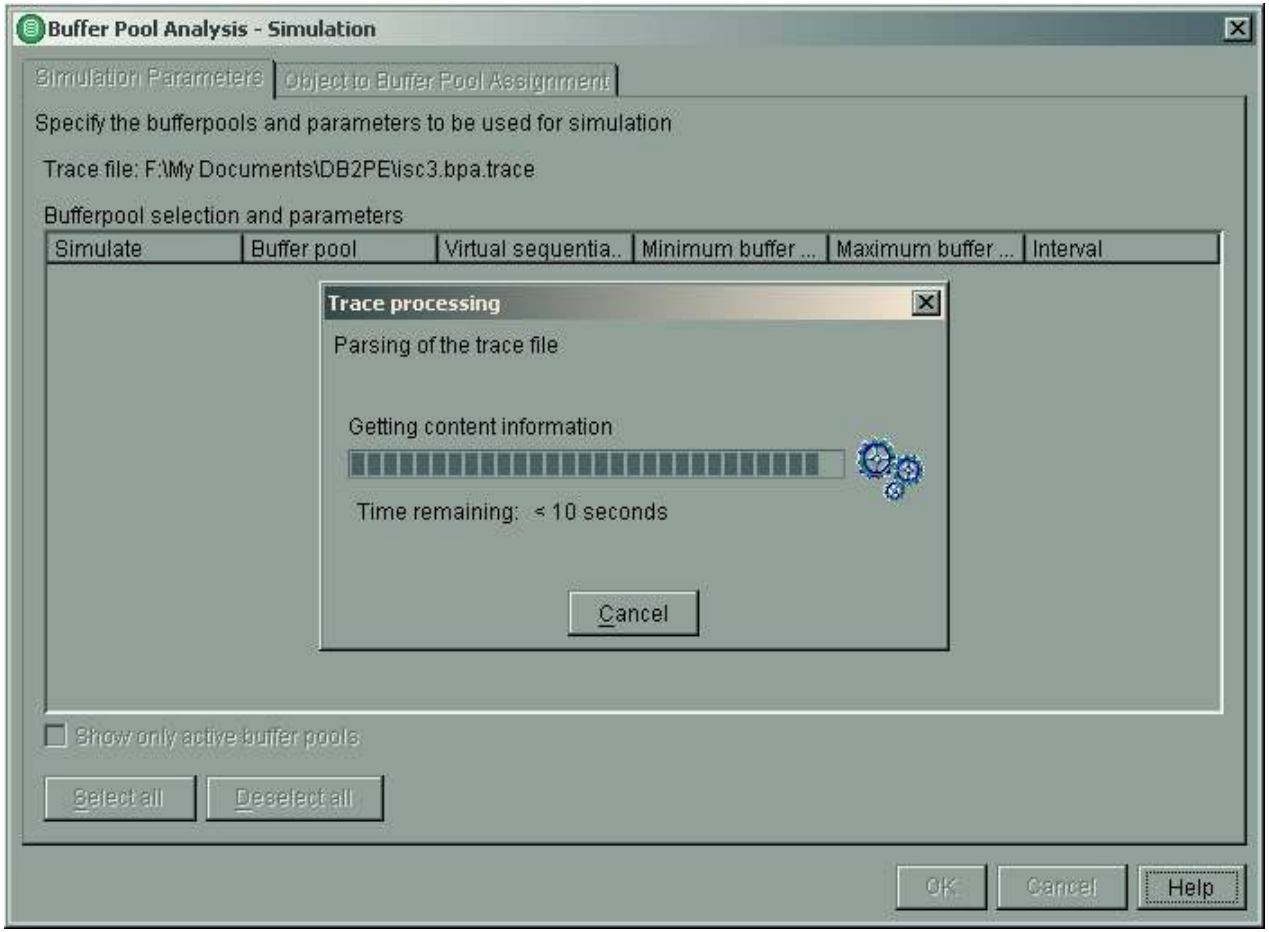
Simulation

- The simulation function can handle trace data files of up to **2 GB**
- The time to preprocess the trace data file and the time to perform a simulation very largely depends on
  - # of active objects in the trace data file
  - # of different buffer pools, and the buffer pool sizes to be simulated
    - For example, on a 2.4 GHz workstation, a 1GB trace data file takes roughly 1.5 minutes to be preprocessed and about five minutes to simulate four buffer pools from 25,000 to 1,000,000 pages (40 sizes)
    - Note that a simulation runs considerably slower if other tasks are using the CPU at the same time

Simulation

- You can adjust the initial simulation parameters as required
  - However, note that the time to perform a simulation increases with the number of different buffer pool sizes to be simulated
- For practical reasons, choose a minimum and maximum buffer pool size and an interval that does not result in more than 40 buffer pool sizes to be simulated
  - Otherwise, the time to generate the simulation result increases excessively. In addition, the simulation result becomes complex and difficult to interpret
- The Virtual sequential threshold field shows the percentage of the virtual buffer pool that might be occupied by sequentially accessed pages, the default value is 80%
- The Simulation buffer pools list shows a set of 50 selectable buffer pools in the range of BP0 to BP49 (the 4 KB buffer pools)
- Up to four selected buffer pools are listed on the left side as Simulation buffer pool #1 to Simulation buffer pool #4

Introduction to BPA - Simulation



- Start a simulation run by selecting ‘File’ -> ‘Start Simulation …’ from the Buffer Pool Analyzer menu bar and open the trace file in the file open dialog
- BPA will then parse the trace and extract the data it needs

- Buffer Pool Analyzer Simulation uses a trace file to calculate the effects of buffer pool changes and object assignment without having to make the changes for real
- It requires a file gathered by Collect Report Data which must be downloaded to the workstation
- The simulation can only include objects which were accessed during data collection

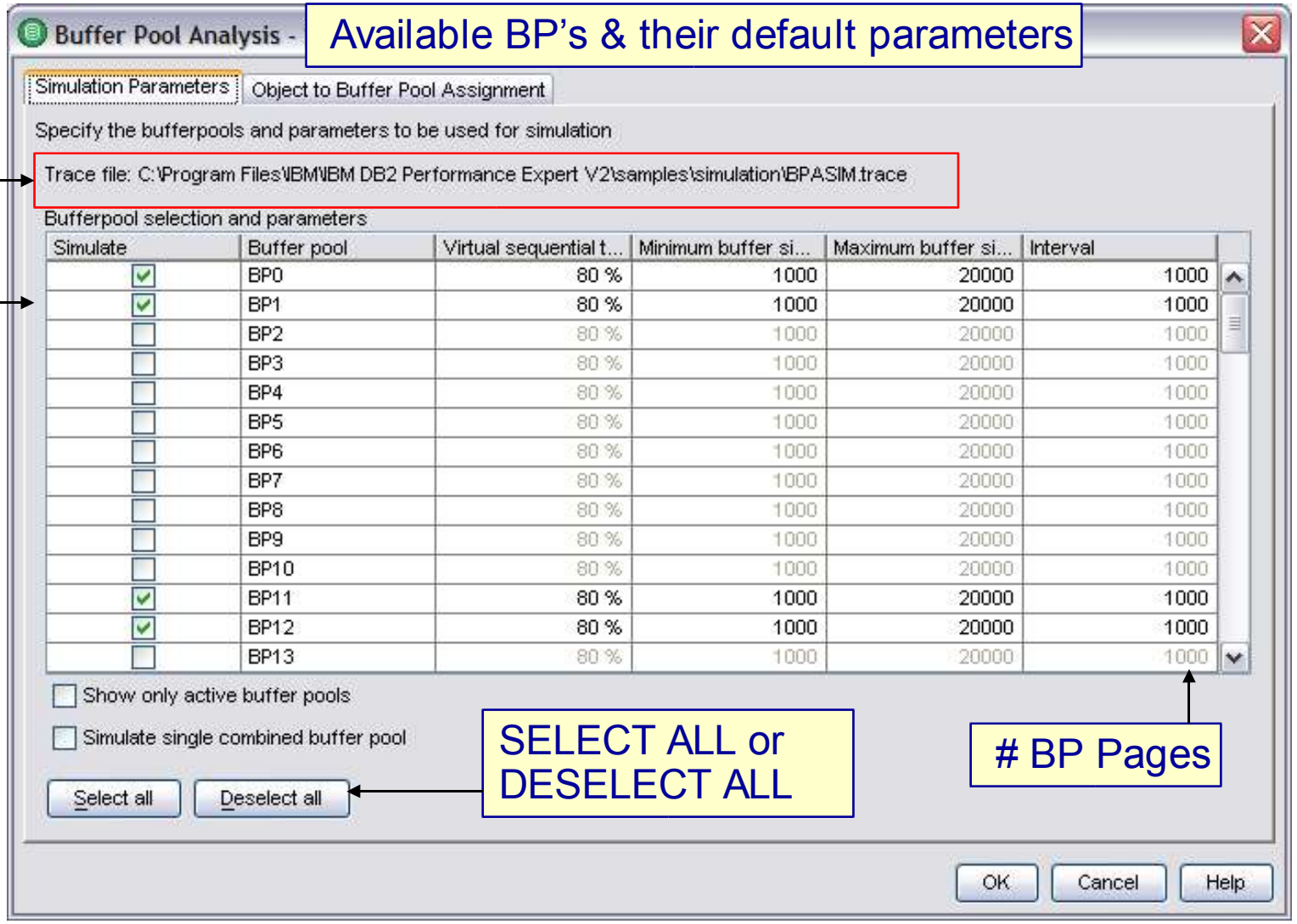
Simulation

Preparing the Simulation Run

Name of TRACE data file

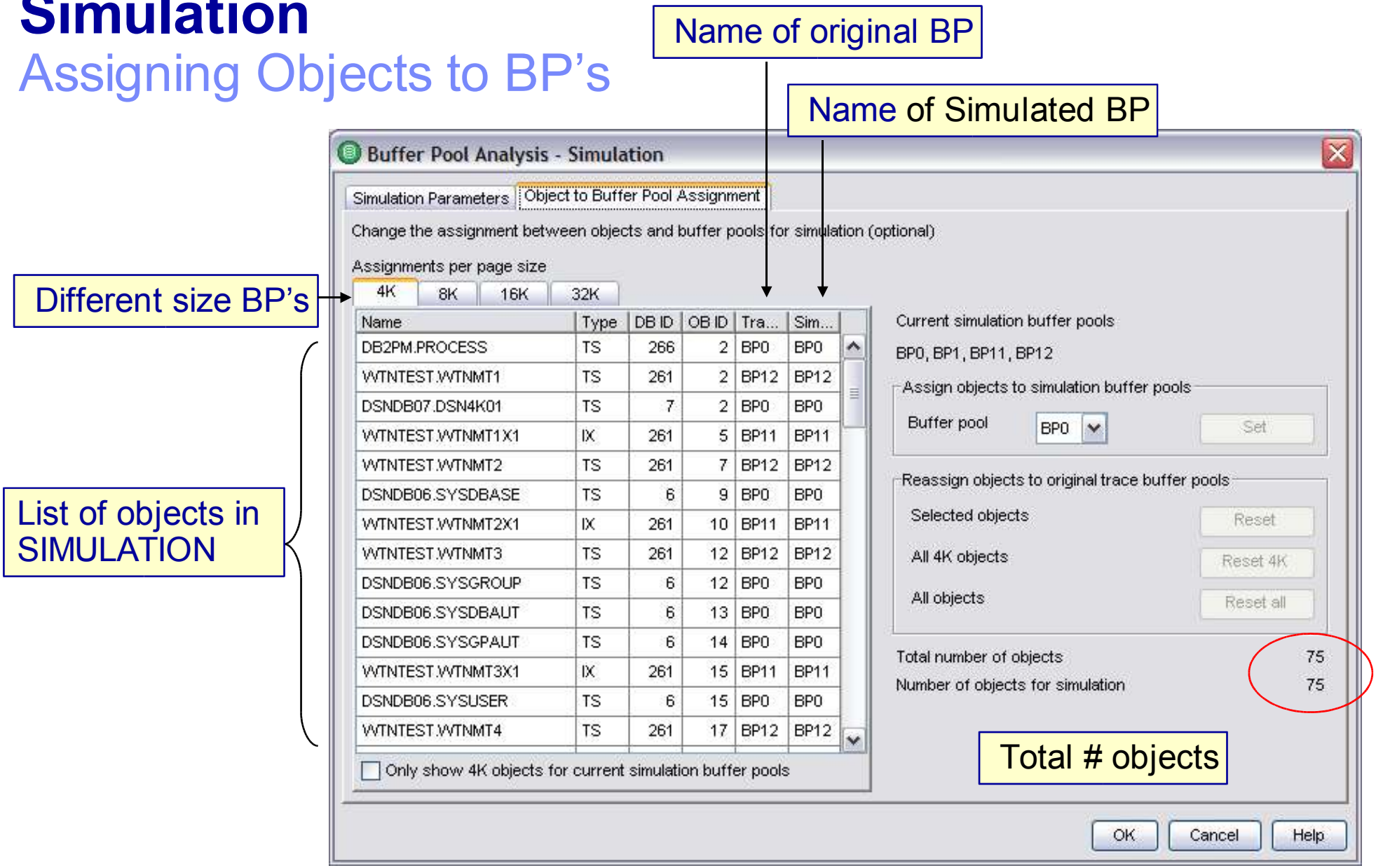
Select buffer pools for simulation and specify characteristics

Active BP's are pre-selected

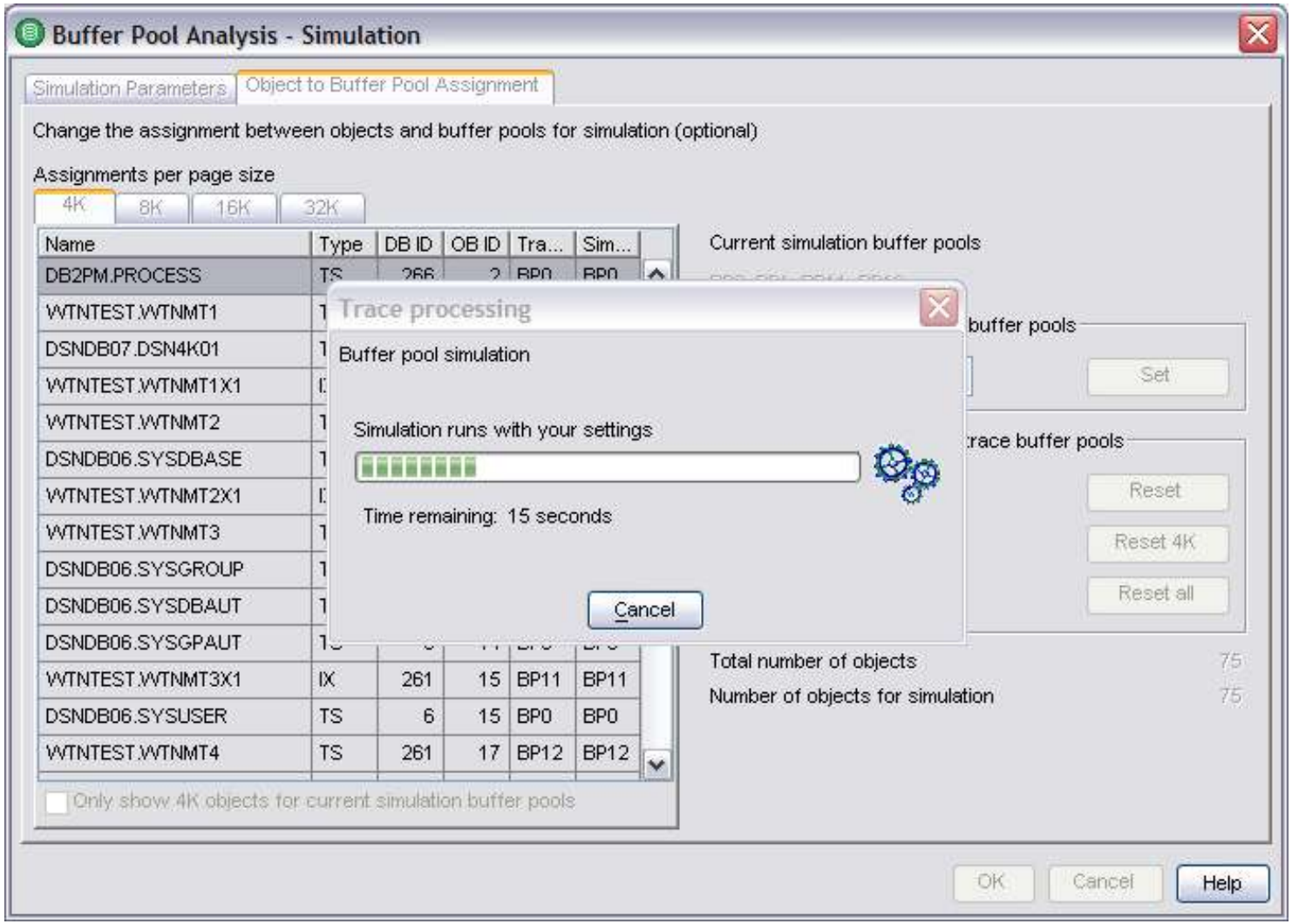


Simulation

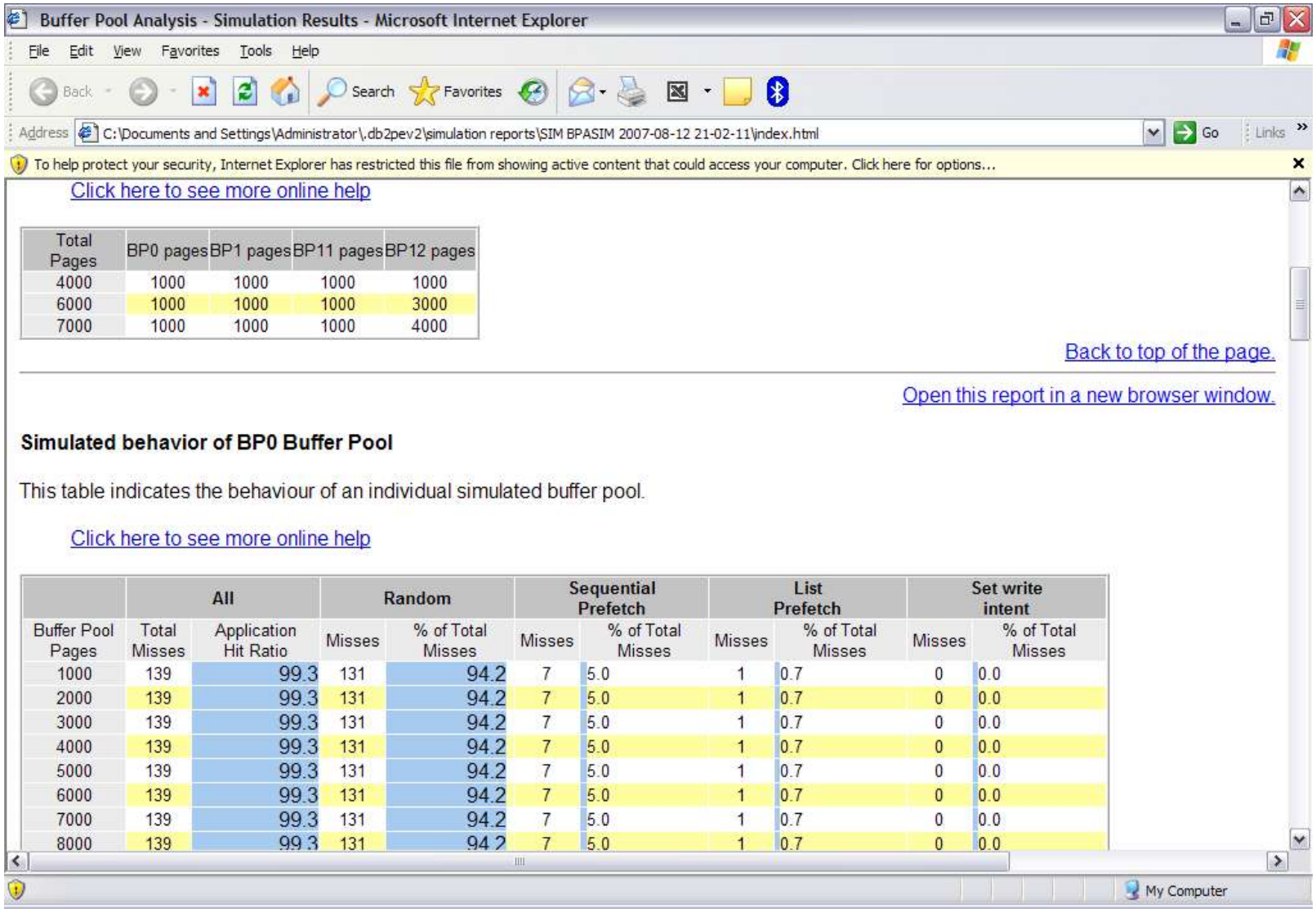
Assigning Objects to BP's



Simulation

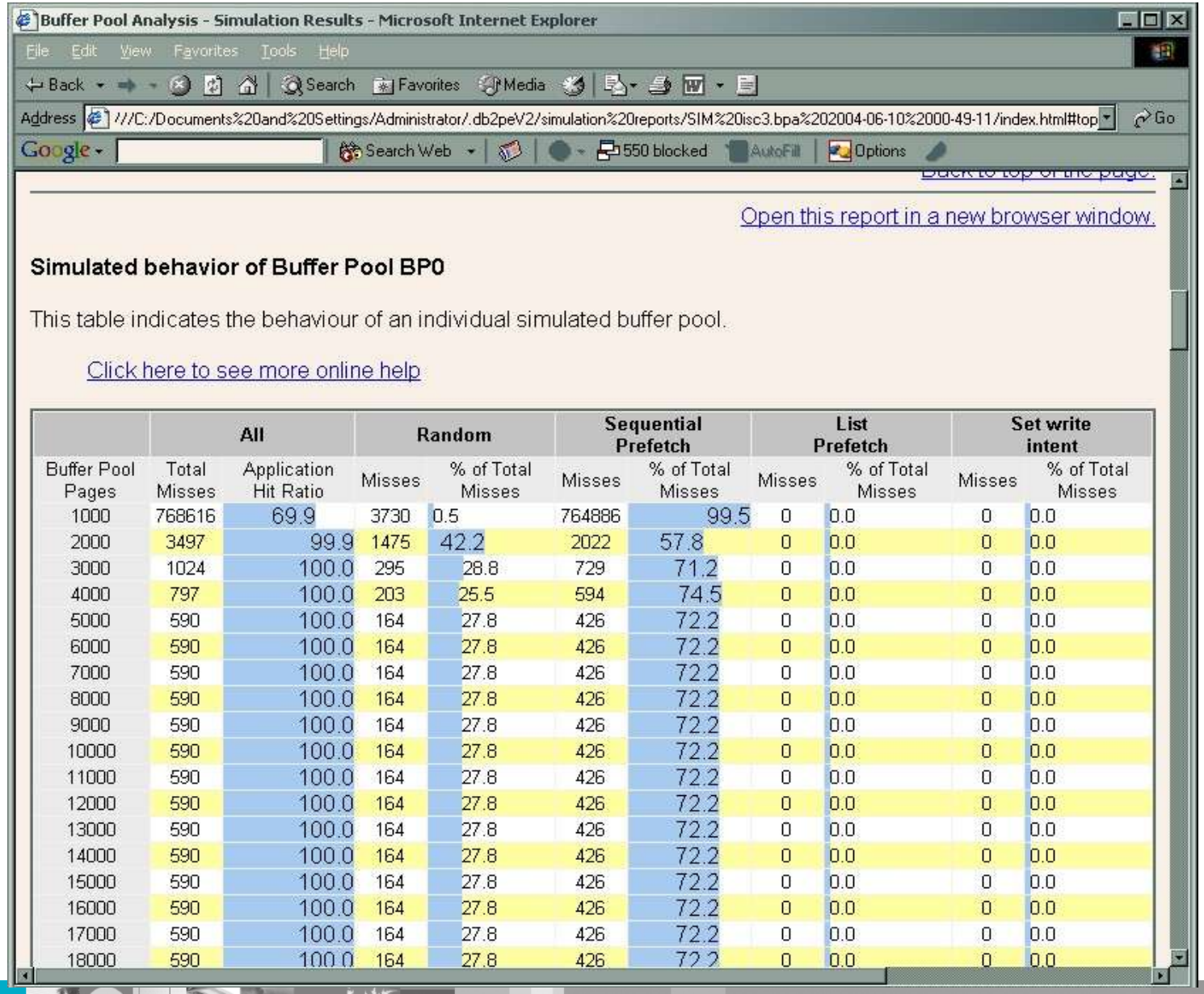

Buffer Pool Analysis - Simulation
Simulation Parameters
Object to Buffer Pool Assignment
Change the assignment between objects and buffer pools for simulation (optional)
Assignments per page size
4K
8K
16K
32K
Name
Type
DB ID
OB ID
Tra...
Sim...
DB2PM.PROCESS
WTNTEST.WTNMT1
DSNDB07.DSN4K01
WTNTEST.WTNMT1X1
WTNTEST.WTNMT2
DSNDB06.SYSDBASE
WTNTEST.WTNMT2X1
WTNTEST.WTNMT3
DSNDB06.SYSGROUP
DSNDB06.SYSDBAUT
DSNDB06.SYSGPAUT
WTNTEST.WTNMT3X1
IX
261
15
BP11
BP11
DSNDB06.SYSUSER
TS
6
15
BP0
BP0
WTNTEST.WTNMT4
TS
261
17
BP12
BP12
Only show 4K objects for current simulation buffer pools
Current simulation buffer pools
buffer pools
Set
race buffer pools
Reset
Reset 4K
Reset all
Total number of objects
75
Number of objects for simulation
75
Trace processing
Buffer pool simulation
Simulation runs with your settings
Time remaining: 15 seconds
Cancel
OK
Cancel
Help

Simulation
Review Reports



Simulation

Review Reports



	All		Random		Sequential Prefetch		List Prefetch		Set write intent	
Buffer Pool Pages	Total Misses	Application Hit Ratio	Misses	% of Total Misses	Misses	% of Total Misses	Misses	% of Total Misses	Misses	% of Total Misses
1000	768616	69.9	3730	0.5	764886	99.5	0	0.0	0	0.0
2000	3497	99.9	1475	42.2	2022	57.8	0	0.0	0	0.0
3000	1024	100.0	295	28.8	729	71.2	0	0.0	0	0.0
4000	797	100.0	203	25.5	594	74.5	0	0.0	0	0.0
5000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
6000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
7000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
8000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
9000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
10000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
11000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
12000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
13000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
14000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
15000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
16000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
17000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0
18000	590	100.0	164	27.8	426	72.2	0	0.0	0	0.0

Simulation Tips

- Result from a simulation shows a comparison of BP efficiency for separate BP's vs. a single combined BP
  - Generally a single BP has the following advantages
    - Improves the total BP hit ratio
    - Requires less monitoring and tuning
    - Treats applications equally
    - Offsets increasing workload of 1 application by decreasing workload in another
  - However a single BP
    - No preference to applications with different levels of importance
    - Different access and usage pattern cannot be isolated

Simulation Tips

- Result from a simulation shows a comparison of BP efficiency for separate BP's vs. a single combined BP
  - Generally multiple BP's have the following advantages
    - Allow for performance preferences
    - Allow for grouping according to access pattern
    - Allow for different thresholds
    - Separate table spaces from indexes
  - Too many BP's increase the effort to monitor and administer BP's

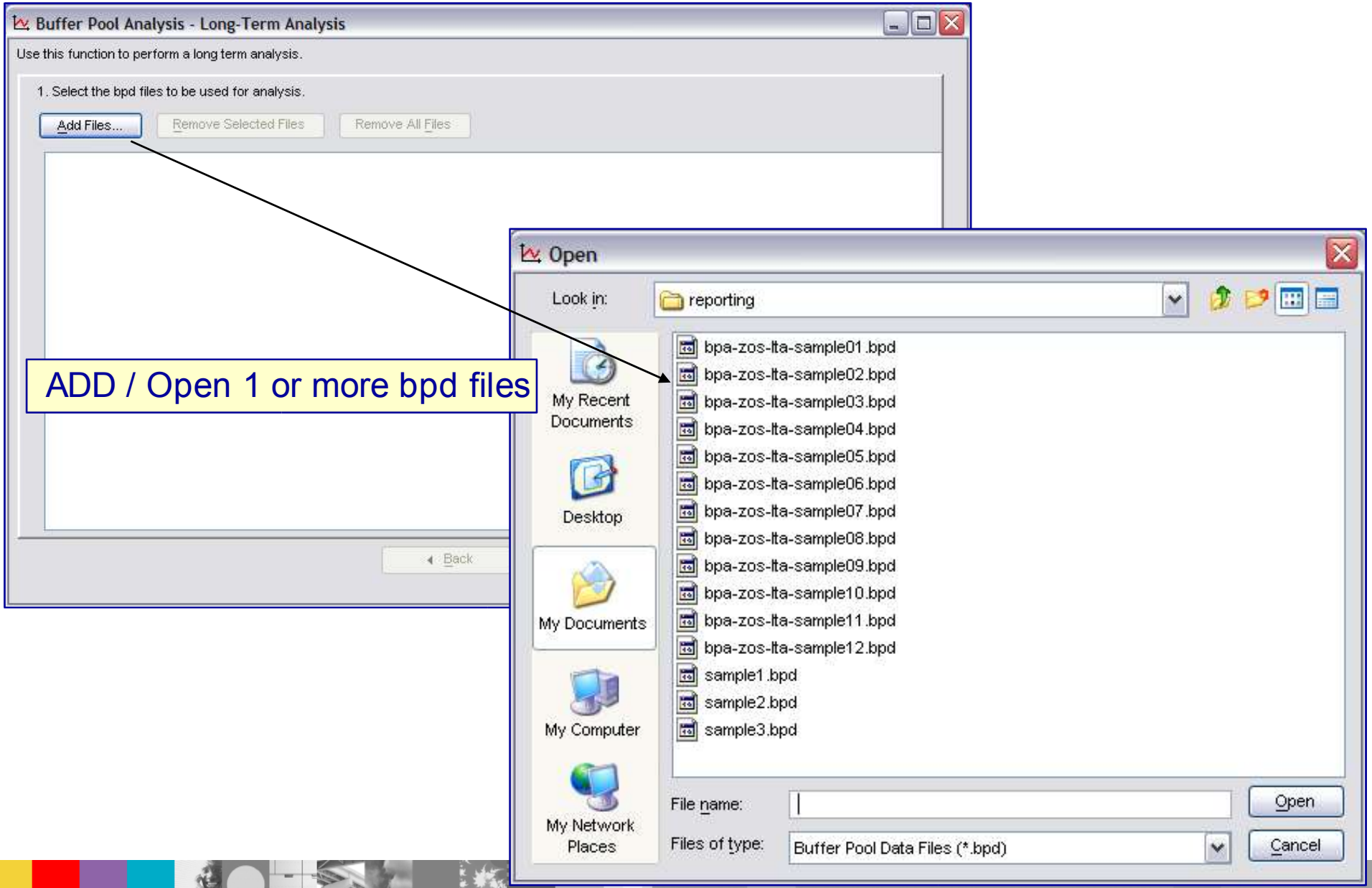
Simulation Tips

- Total BP size has a great effect on performance
  - If it is too large, and there is not enough memory to allocate them, then a minimum BP for each page size will be allocated and performance will be reduced
- Smaller BP's are more likely affected by fluctuating workload
- Smaller BP's tend to show more performance peaks
- Size of BP's that predominantly process write requests can be minimized
  - Usually show a low hit ratio
  - Consecutive write operations fill up the BP and require the data to be frequently written to disk

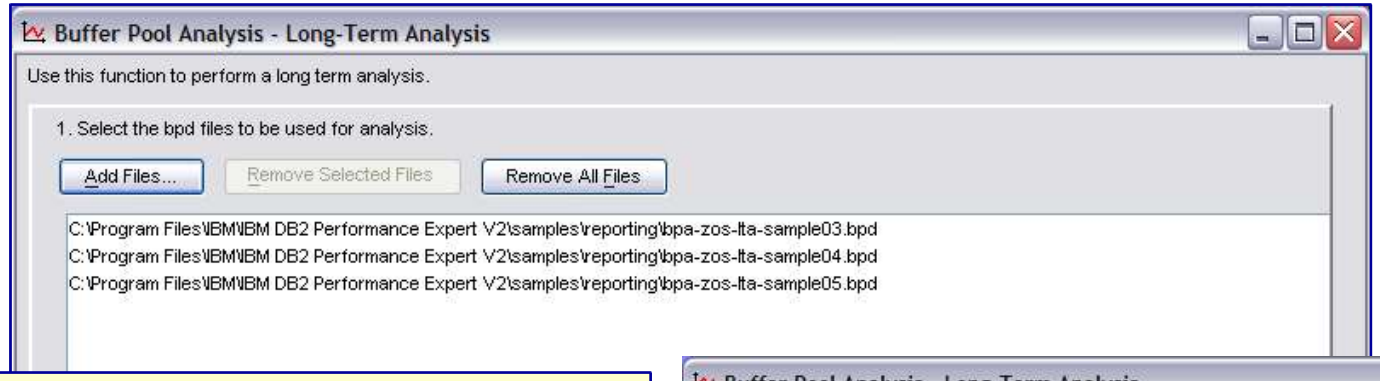
Long-Term Analysis

- ☑ Collect statistical information
  - CRD
  - Batch job - hlq.TKO2SAMP(BPOMACRD)
- ☑ FTP trace file to client
  - BIN
  - .bpd extension
- ❑ Bring up the GUI interface
  - ❑ Click on TOOLS and select Buffer Pool Analyzer
  - ❑ Click on FILE and select Long-Term Analysis
  - ❑ Select a .bpd file

Long-Term Analysis

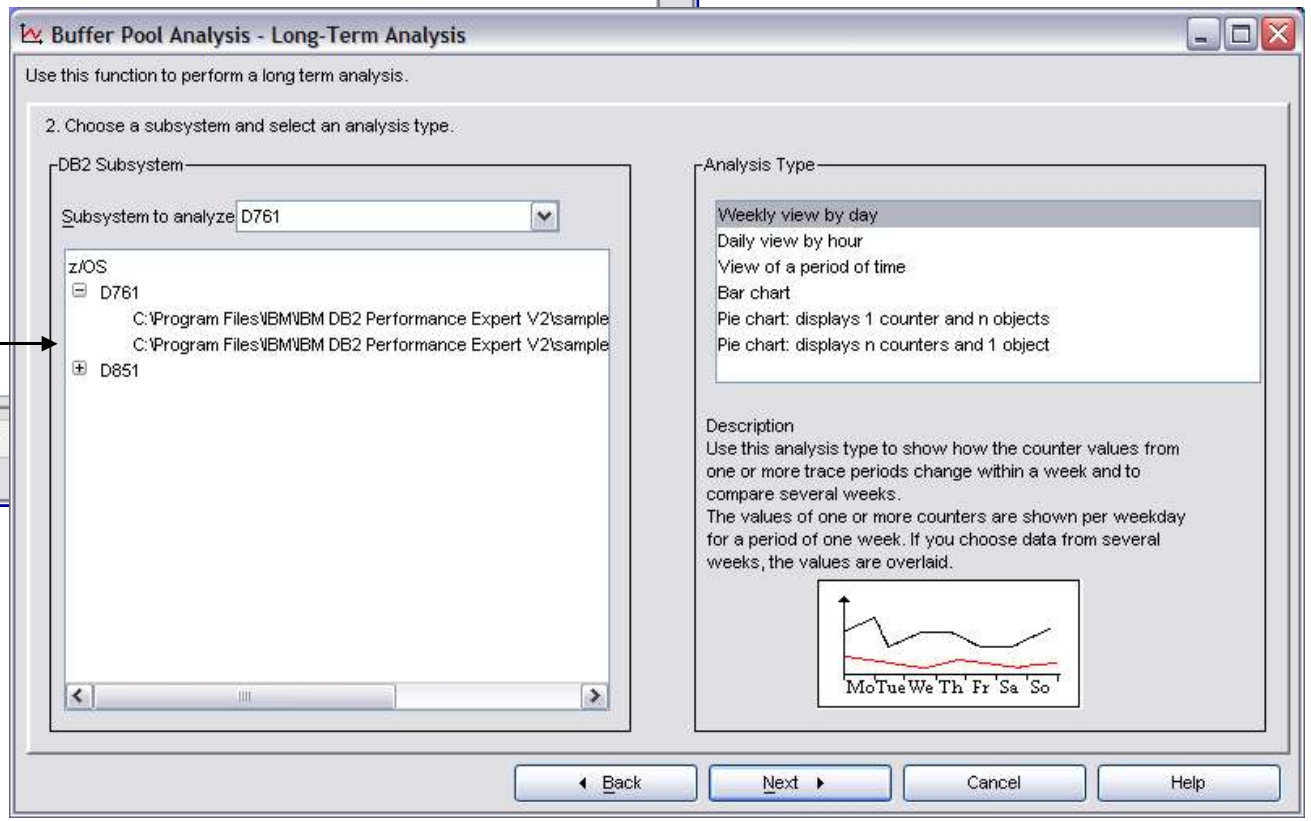


Long-Term Analysis

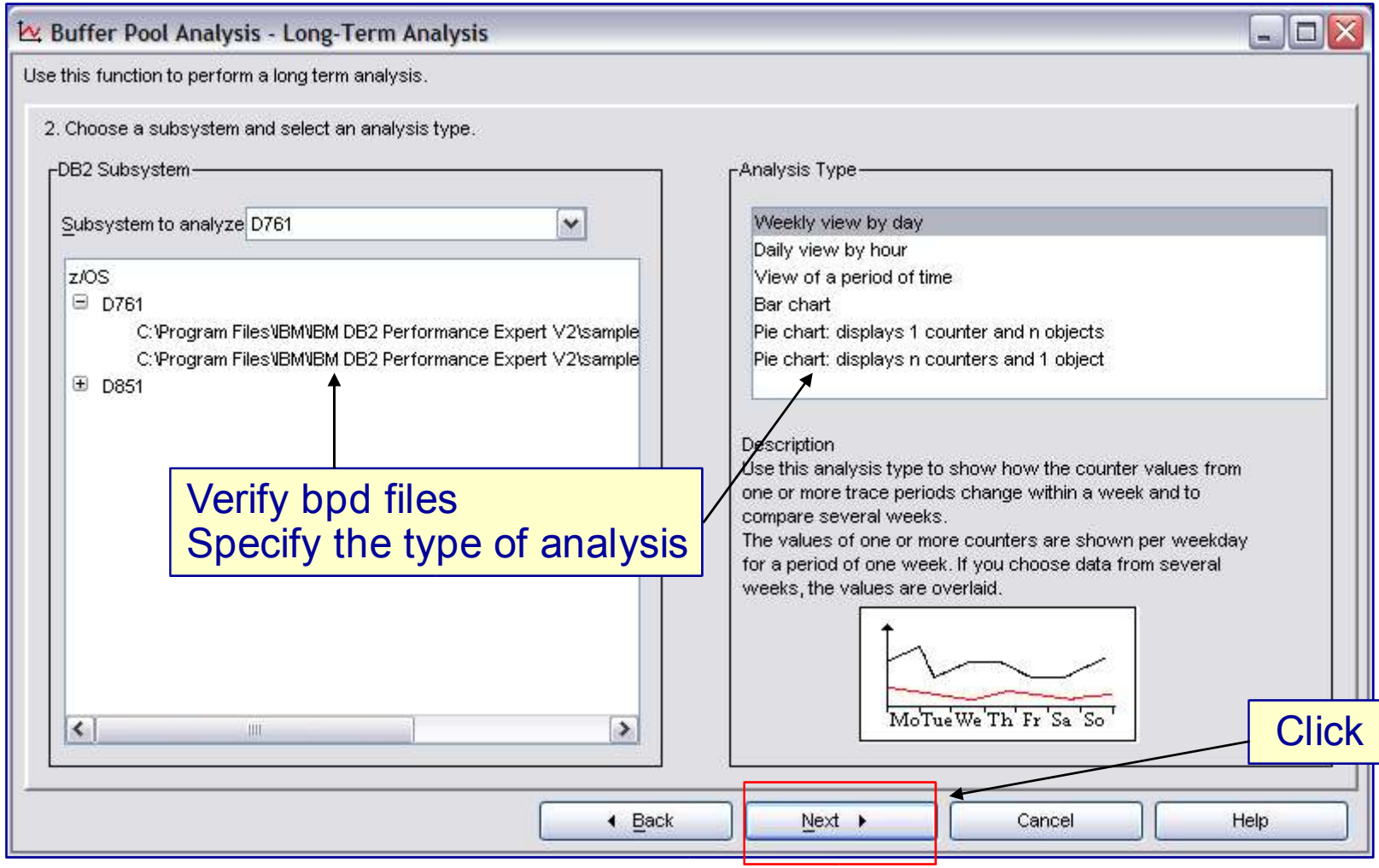


Click on NEXT

1. bpd files are opened
2. Data is pre-processed
   - Subsytems are determined
   - Counters and active objects are determined
   - Earliest and latest timestamp in ea. bpd file is determined



Long-Term Analysis



A meaningful Long Term Analysis can only be performed with data form 1 subsystem

Analysis Types

- **Weekly View by Day**
  - Shows counter values per weekday of selected counters and objects
  - Used to
    - Analyze how certain counters develop over a week
    - Compare how counters develop over several weeks
  - Looking for conspicuously high or low values
  - Counter values = per minute values
  - One counter value / counter, object, and weekday is shows
    - Ex. 5000 Getpage total operations / minute on avg. over a day
  - Counter values are connected by lines for better readability

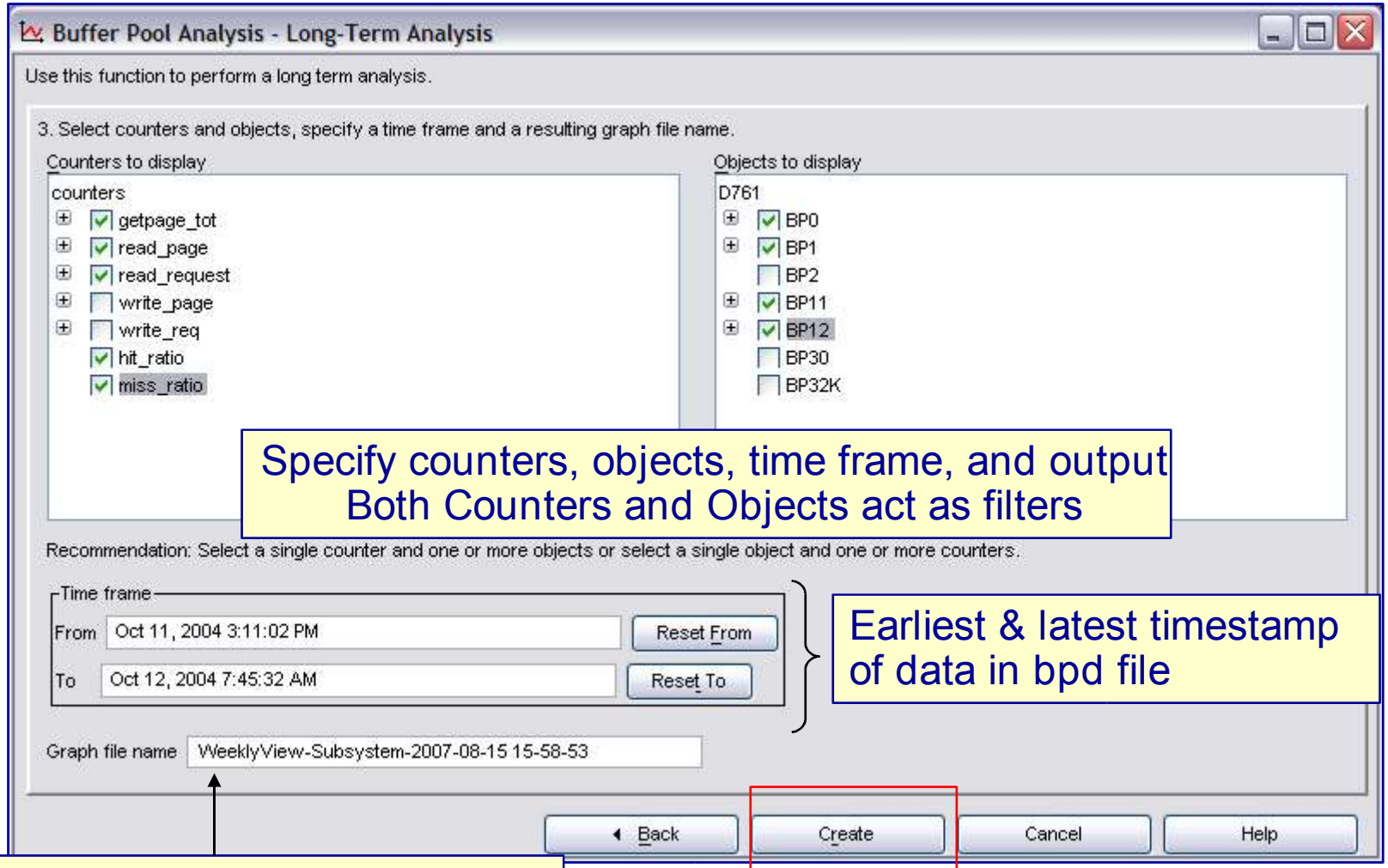
Analysis Types

- **Daily View by Hour**
  - Shows counter values per hour of selected counters and objects
  - More detailed analysis than the Weekly View by Day
  - Used to
    - Analyze how certain counters develop over a day
    - Compare how counters develop over several days
  - Counter values = per minute values
- **View of a Period of Time**
  - Shows counter values of selected counters and objects from several bpd files in chronological order
  - Counters represent per minute values

Analysis Types

- **Bar Chart**
  - Shows the distribution of counter values of selected counters / objects via a bar chart
  - Counters are per minute values
  - Use to compare selected counters in selected objects
- **Pie Chart Display 1 Counter and n objects**
  - Each slice accepts one of the selected objects
  - Use to compare a few values to a total (ex. How much of the GETPAGE total activity happens in the most important BP's)
- **Pie Chard Display n Counters and 1 Object Analysis**
  - Each slice represents 1 of the selected counters
  - Compare a few values to a total (ex. Determine which counters have the most activity in a BP)

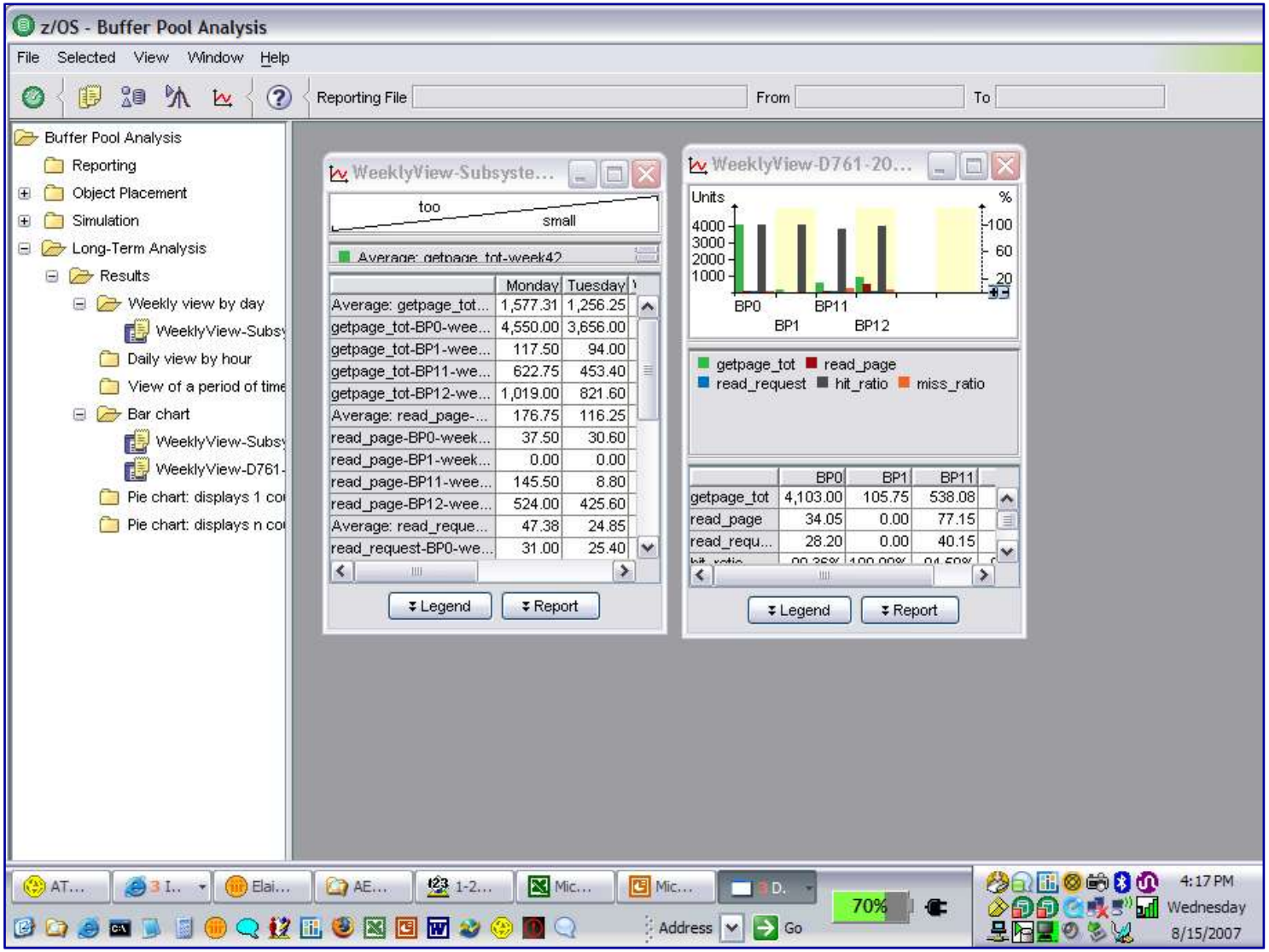
Long-Term Analysis



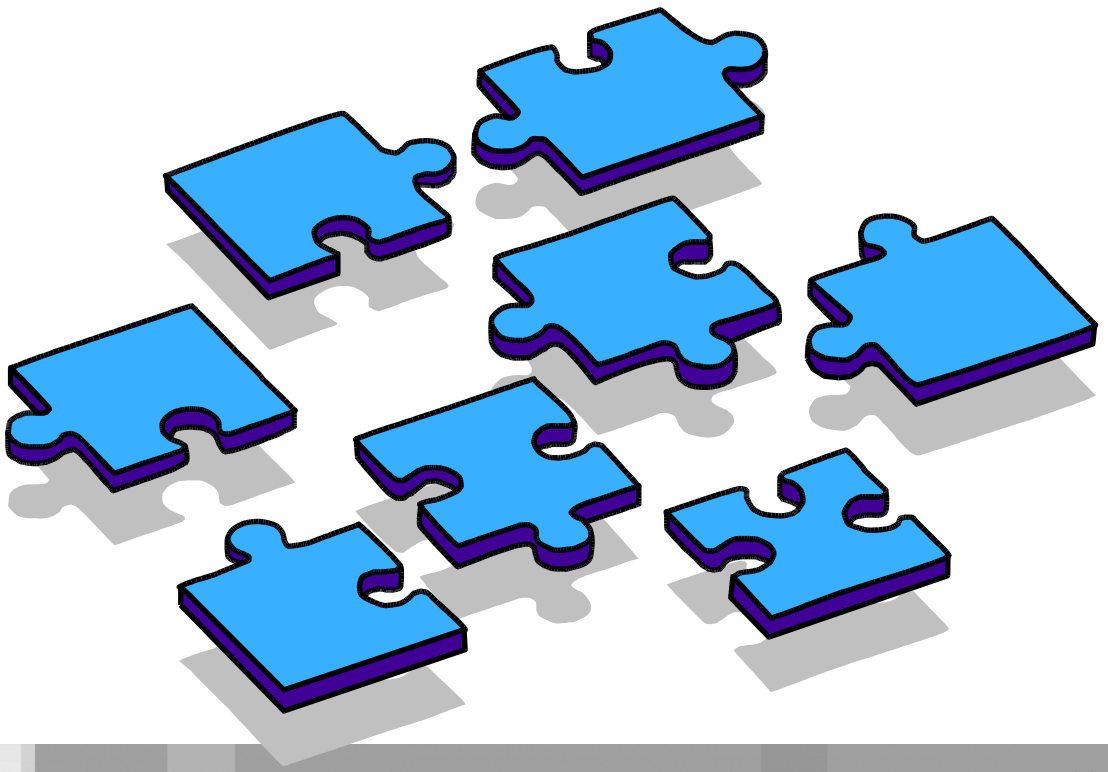
Long-Term Analysis

- See how a single counter behaves in several objects (compare GETPAGE sequential counts of several objects)
  - Select one counter
  - Select several objects
- See how several counters behave in 1 object (analyze key counters on an object
  - Select the counters
  - Select a single object

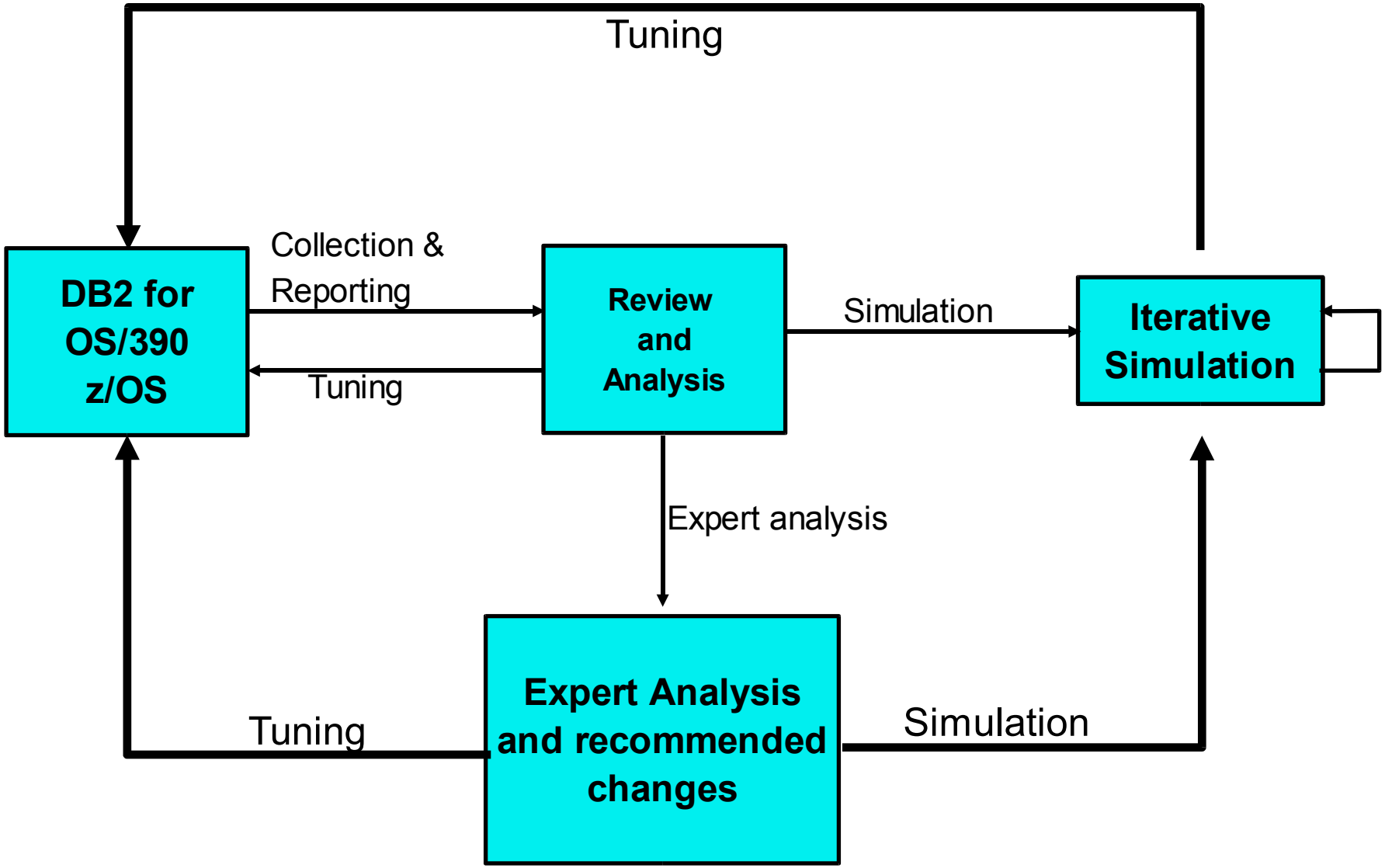
Long-Term Analysis



Putting It All Together



Buffer Pool Analyzer - A Logical View



One Approach
Using Batch Reports

- Start with **summary report**
  - Data is collected over 30 minutes, with **short record format**
  - The **trace data file** is used for **activity reports** and the **bpd file** generation
  - The trace data file and the bpd file are used in later steps for object placements and simulations on the workstation
- **Analyze** the summary report
  - The summary report, ordered by BPID-QPAGESET and sorted by ASYNCPAGE
  - Look for anomalies (some examples):
    - Hit ratios
    - Applications performing lots of random getpages
    - Dynamic prefetch activity
    - Ratio of Dynamic prefetch to sequential prefetch (indicating possible need for application change - move away from random scanning of tables or indexes to sequential access)

Using Batch Reports

- **Generate Detail Report**
  - Ordered by BPID-QPAGESET and sorted by BPID and GETPAGE will show the most active objects
    - Look for concentrations of activity in a few objects
    - Look for objects with high percentage of misses
  - Ordered by BPID-QPAGESET and sorted by BPID and READSYNC shows the most “expensive ”objects in terms of I/O
    - Shows most "expensive" objects by I/O
    - Review synchronous and average delay times

Report information may indicate that ‘simulating’ the effect of moving objects into the own BP may be beneficial

Another Approach
Analyze Effects of Different BP Attributes

- Perform a **simulation** using the actual object placements:
  - Choose the four most active buffer pools as shown by the **activity reports**
  - Using the existing buffer pool sizes as a midpoint, select minimum and maximum page size for all buffer pools that results in no more than 40 sizes
    - For example, if the actual BP size is 10000 pages, pick 5000 as a start, 20000 as an upper bound, and 375 pages (20000 - 5000 / 40) as the increment
  - Don't change any buffer pool assignments
    - Review the "Comparing separate Buffer Pools versus a single combined Buffer Pool" for possibly combining pools
    - Review the "Recommended sizing for separate Buffer Pools" reports to see recommended memory allocation
    - Review the "Simulated behavior of each separate Buffer Pool " for change in miss rate as sizes increase to determine best benefit
    - The other tables in this simulation result are only required for very detailed analysis, especially for application tuning

ANOTHER APPROACH
Analyze Effects of Different BP Attributes

- Perform an **object placement analysis**
  - Use default rule set as suggested by BPA
  - Don't change any placement
- Review the object assignment recommendations from object placement
- Perform a second simulation with results from object placement (assuming that one or more object placements to different buffer pools are recommended)
  - Review the “Simulated behavior of each separate Buffer Pool ”
    - Look for favorable changes in application hit ratios and reductions in misses
- Use additional object placement analysis to obtain recommendations for splitting like objects; use simulation to help understand effect on hit ratios

Conclusion

- There a many opportunities to improve DB2 System and application performance
- While physical design and SQL tuning provide great paybacks, these efforts may be very work intensive
- One of the best ways to improve performance is through Buffer Pool Tuning
- The Buffer Pool Analyzer makes this a scientific process taking the guess work out of the tuning

- SC18-9986 Buffer Pool Analyzer User's Guide